Task Facilitator

PROJECT REPORT

ON

TASK FACILITATOR

BY

1) SAKSHI KANKUSE

2) ASHISH JOSHI

3) NARGIS SHAIKH

4) SURAJ WATHORE

5) VIVEKANAND HUDGE

SAVITRIBAI PHULE

PUNE UNIVERSITY

MASTER OF COMPUTER APPLICATION

ASM'S IBMR

PUNE-411019

2021-23

INDEX

CHAPTER 1. INTRODUCTION

Task Facilitator is a software application for maintaining a user detail's. This is a client based desktop application system, to computerize at user work. This project currently includes modules like

- Add User.
- View User.
- Delete User.
- Assign Task to User.
- View Task.

1.1) EXISTING SYSTEM:

The existing system is majorly clerical process this is not computerized system. There are some drawbacks in existing system:

• The existing system performs the task as manually.

• In this system there are lots of paper works.

• The existing system is very time consuming processes.

• It is Difficult to generate reports.

• There is difficulty in storing and retrieving the data.

• It has a large amount of repeated data.

• There is lots of man power required.

Due to the above drawback, the existing system is very complicated and not secure system.

1.2) NEED OF THE SYSTEM:

- Now our system will overcome this all drawbacks.
- It will reduce efforts required to manage all task records efficiently.
- All work can be done on just few clicks.
- The Only need is to fill given forms for retrieving required information.
- This system will provides facilities like add user records, view user records, delete user records , view & update tasks records.
- The system helps to maintain users and tasks which are required for admin for a sustainable business lifecycle.

1.3) SCOPE OF THE SYSTEM:

Provide the information related to the System & add new user's and Assign numerous Task to user according to business needs.

- Admin:

  - Add user- Admin has total control on admin section which consists of modules through which admin has the authority to keep a record of user details.
  - View user- Only admin can access this module & admin has ability to view the registered users which were added in previous module.
  - Delete user- Admin has this feature through which he could easily perform deletion of user records through users respective ID's from the existing user details .

- Assign task- This is one of the crucial feature on the admins section which allows admin to appoint a user to perform a task , it will only assign the task successfully if the user id is present in the stored user records.
- Completed tasks- This module focuses on storing records of the numerous tasks which are completed, records present in this module are added when user's update the task from user section through their respective ID's beside that through this module admin can generate reports with ease.

➢ <u>User:</u>

- View task- This module is a part of user's section & displays when a user logs in successfully which was earlier registered by admin, primary goal of this module is to view the task which admin has assigned to users, here all the assigned tasks can be viewed by users and they are in progress state.
- Update task- This module plays a vital role on users section as it allows users to update the status of their current task from process to completed through their respective ID's once the user has updated their task admin will be able to view these tasks in Completed task on Admin section.

1.4) OPERATING ENVIRONMENT HARDWARE AND SOFTWARE:

Hardware/Software Requirements:-

- **CLIENT SIDE:**

Operating System	Windows 10
Processor	Intel-Pentium Dual Core
RAM	1 GB RAM
Browser	Internet Explorer, Google Chrome, Mozilla firefox, etc.

- **SERVER SIDE:**

Operating system	Windows 10
Front End	Java,Swing
Connectivity	JDBC Drivers
Backend	MySQL Database Server

1.4) DETAIL DESCRIPTION OF TECHNLOGY USED:

- JAVA:
  - Java is a general-purpose, object-oriented programming language developed by Sun Microsystems of USA in 1991.Originally called Oak by James Gosling (one of the inventor of the language).
  - Java was invented for the development of software for consumer electronic devices like TVs etc.
  - The main aim had to make java simple, portable and reliable.
  - Java Authors: James , Arthur Van , and others. Java is a high-level, third generation programming language, like C, FORTRAN, Smalltalk, Perl, and many others. You can use Java to write computer applications that play games, store data or do any of the thousands of other things computer software can do.
  - Compared to other programming languages, Java is most similar to C. However although Java shares much of C's syntax, it is not C. Knowing how to program in C or, better yet, C++, will certainly help you to learn Java more quickly, but you don't need to know C to learn Java.
  - A Java compiler won't compile C code, and most large C programs need to be changed substantially before they can become Java programs. What's most special about Java in relation to other programming languages is that it lets you write special programs called applets ,web project etc. that can be downloaded from the Internet and played safely within a web browser. Java language is called as an Object-Oriented Programming language and before

beginning for Java, we have to learn the concept of OOPs(Object-Oriented Programming).

- **SWING:**

  - Swing is a lightweight Java graphical user interface (GUI) that is used to create various applications.
  - Swing has platform-independent components. It enables the user to create buttons and scroll bars.
  - Swing includes packages for creating desktop applications in Java. Swing components are written in Java language. It is a part of Java Foundation Classes (JFC).

- **MYSQL:**

  - **MySQL** ("My S-Q-L", officially, but also called "My Sequel") is (as of July 2013) the world's second most widely used open-source relational database management system (RDBMS).
  - It is named after co-founder Michael Widenius daughter, My. The SQL phrase stands for Structured Query Language. The MySQL development project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements.
  - MySQL was owned and sponsored by a single for-profit firm, the Swedish company MySQL AB, now owned by Oracle Corporation. MySQL is a popular choice of database for use in web applications, and is a central component of the widely used LAMP open source web application software stack (and other 'AMP' stacks). LAMP is an acronym for "Linux,

Apache, MySQL, Perl/PHP/Python." Free-software-open source projects that require afull-featured database management system often use MySQL.

- For commercial use, several paid editions are available, and offer additional functionality.
- Applications which use MySQL databases include: TYPO3, MODx, Joomla, WordPress, phpBB, MyBB , Drupal and other software . MySQL is also used in many high profile , large-scale websites, including Wikipedia, Google (though not for searches), Facebook, Twitter, Flickr, and YouTube.

CHAPTER 2. PROPOSED SYSTEM:

2.1) PROPOSED SYSTEM:

The aim of proposed system is to develop a system of improved facilities. The proposed system can overcome all the limitations of the existing system. The system provides proper security and reduces the manual work.

- Security of data
- Ensure data accuracy
- Proper control of the higher officials.
- Minimize manual data entry.
- Minimum time needed for the various processing.
- Greater efficiency.
- Better service.
- User friendliness and interactive.
- Minimum time required
- Fast access to database
- Less error
- More storage capacity.
- Search facility

2.2) OBJECTIVES OF THE SYSTEM:

Task Facilitator system is a simple project which make easy to interact with user. This project will allow the admin to manage the user information with ease. This system allows us to add, delete, view user's details for the future & assign task to the user get easier and faster.

2.3) USER REQUIREMENTS:

Functional Requirement

On the other side, there are those that deal with all type of technical functioning of the system.

- **Admin Login:**

He/she is to authenticate a user, that is to know whether he or she can get access to the system. At the time of login, the user will be required to enter their user id and password. If for any user these fields don't match, then the user will not be allowed to use the system. For this, the user id is stored at the time of registration.

This task facilitator system must only allow a user with a valid id and password to become the login as Admin. After this authorization takes place, to know all are the levels a particular user can access to.

- **Adding New User:**

Admin has the authority or access to add new user. The system must enter and maintain the number of Details of each individual User. Also, the system must allocate unique IDs to individual User carefully.

- **View User:**

The system must provide the facility of searching on their unique id, the name of the user and other details. Some table views of the searches must be available.

Non-Functional Requirement of Library Management System

- **Product Requirements**

These are those that specify some criteria that can be used to evaluate the performance of a system in some particular conditions.

- **Efficiency Requirement:**

Through this system, the members and the librarian gets a way to ease their work. Through this system, the member can search and get the book issued easily. Also, less time will be needed to spend by the librarian to handle this. Therefore, the throughout is faster processing of the library management system.

- **Reliability Requirement:**

The system does its work with more accuracy like user registration to the system, user validation, and authorization, book search, and issue operation return status, and updating the database by synchronizing between database and application.

- **Usability Requirement:**

The proposed library management system provides a user-friendly environment to the users so that the librarians, as well as the students, can utilize the system in an effective manner for ease of work.

- **Portability requirements:**

Portability in high-level computer programming is the usability of the same software in different environments. The pre-requirement for portability is the generalized abstraction between the application logic and system interfaces. When software with the same functionality is produced for several computing platforms, portability is the key issue for development cost reduction.

Transferring installed program files to another computer of basically the same architecture. Reinstalling a program from distribution files on another computer of basically the same architecture.

- **Organisational Requirements**

- **Delivery Requirement:**

There is always some time duration specified to develop a project. Similarly, this system is expected to be complete within 2 months of time. This launch will be used for improving the performance, as it will be evaluated by the users and then the problems that are occurring with the system will be solved.

- **Implementation requirements:**

Implementation is the realization of an application, or execution of a plan, idea, model, design, specification, standard, algorithm, or policy. an implementation is a realization of a technical specification or algorithm as a program, software component, or other computer system through programming and deployment. Many implementations may exist for a given specification or standard. In implementing whole system. it uses html in front end with java awt and swing. scripting language which will be used for database connectivity and the backend ie the database part is developed using mysql.

- **Standard requirements:**

The project should be developed as per standard format specified by IEEE. Typical platforms include a computer architecture, operating system, programming languages and related user interface. The product should be developed as per client's standard requirements.

❖ **External Requirements**

- **Interoperability requirements:**

Interoperability is a property of a product or system, whose interfaces are completely understood, to work with other products or systems, present or future, without any restricted access or implementation. The IEEE Glossary defines interoperability as: the ability of two or more systems or components to exchange information and to use the information that has been exchanged.

- **Legislative requirements:**

In the proprietary software industry, an end-user license agreement or software license agreement is the contract between the licensor and purchaser, establishing the purchaser's right to use the software. The license may define

ways under which the copy can be used. Software companies often make special agreements with large businesses and government entities that include support contracts and specially drafted warranties.

- **Privacy requirements:**

The term "privacy" means many things in different contexts. Different people, cultures, and nations have a wide variety of expectations about how much privacy a person is entitled to or what constitutes an invasion of privacy. Privacy is the ability of an individual or group to seclude themselves or information about themselves and thereby reveal themselves selectively. The boundaries and content of what is considered private differ among cultures and individuals, but share basic common themes. Privacy is sometimes related to anonymity, the wish to remain unnoticed or unidentified in the public realm.

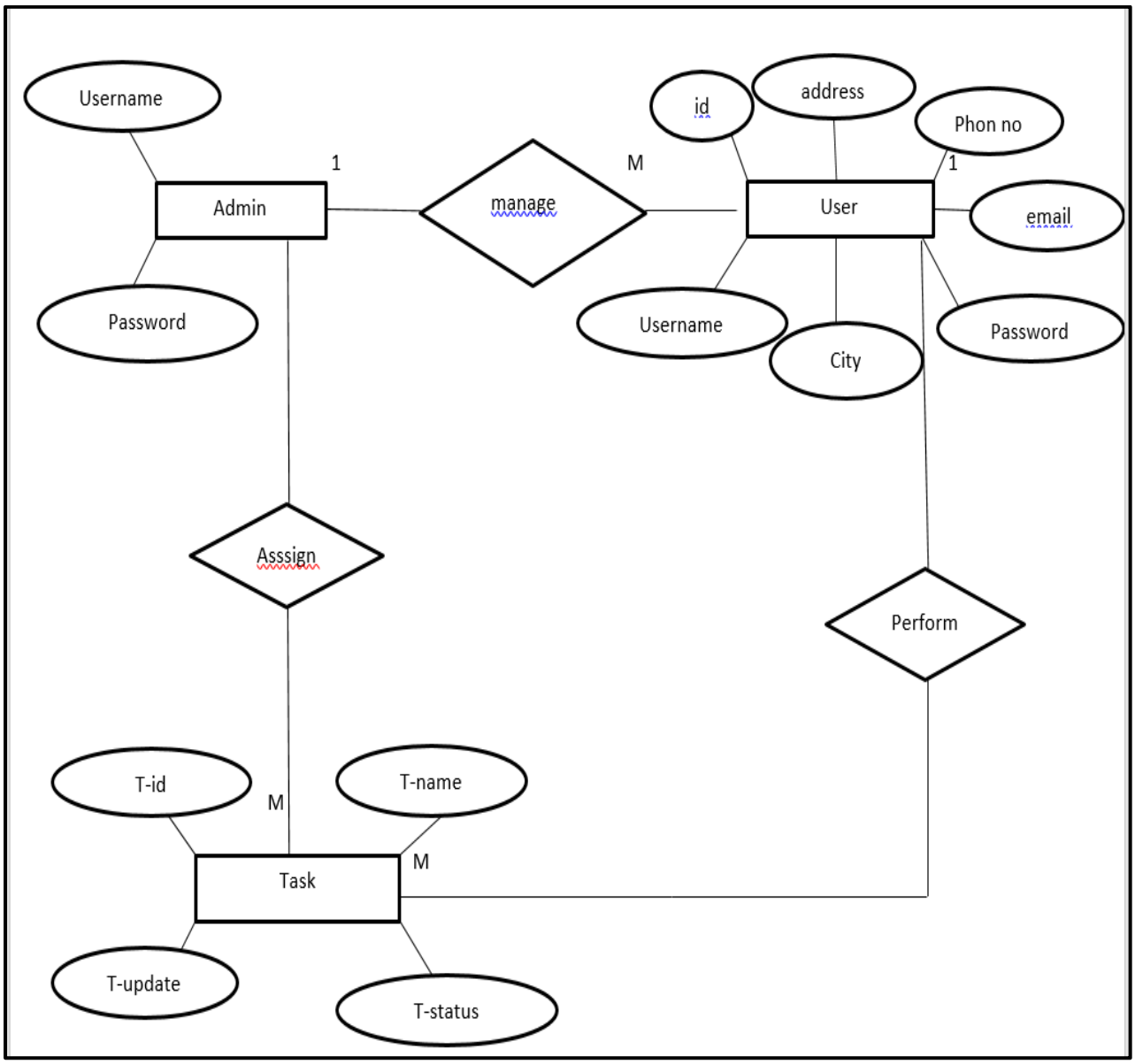
- **Safety requirements:**

Safety can also be defined to be the control of recognized hazards to achieve an acceptable level of risk. Safety is the state of being "safe", the condition of being protected against physical, social, spiritual, financial, political, emotional, occupational, psychological, educational or other types or consequences of failure, damage, error, accidents, harm or any other event which could be considered non-desirable.

;

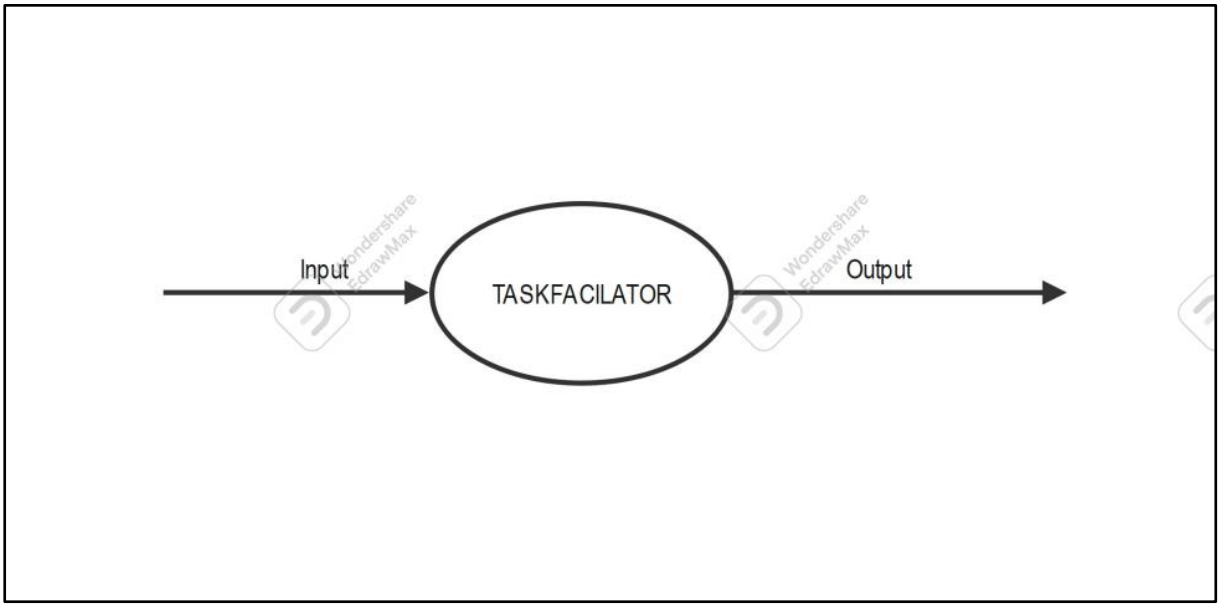
CHAPTER 3) ANALYSIS AND DESIGN

3.1) ENTITY RELATIONSHIP DIAGRAM:

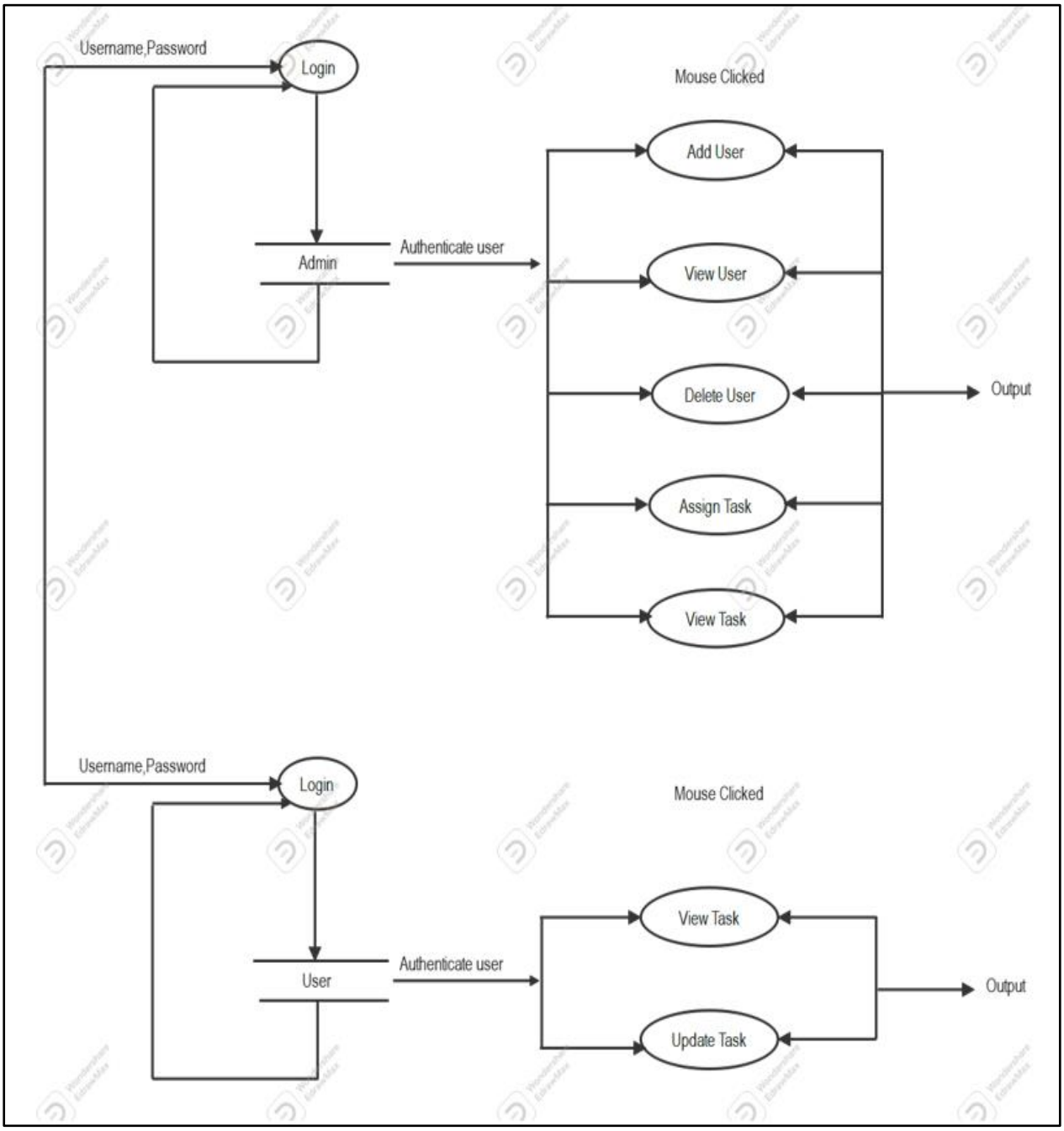


3.2) DATA FLOW DIAGRAM:

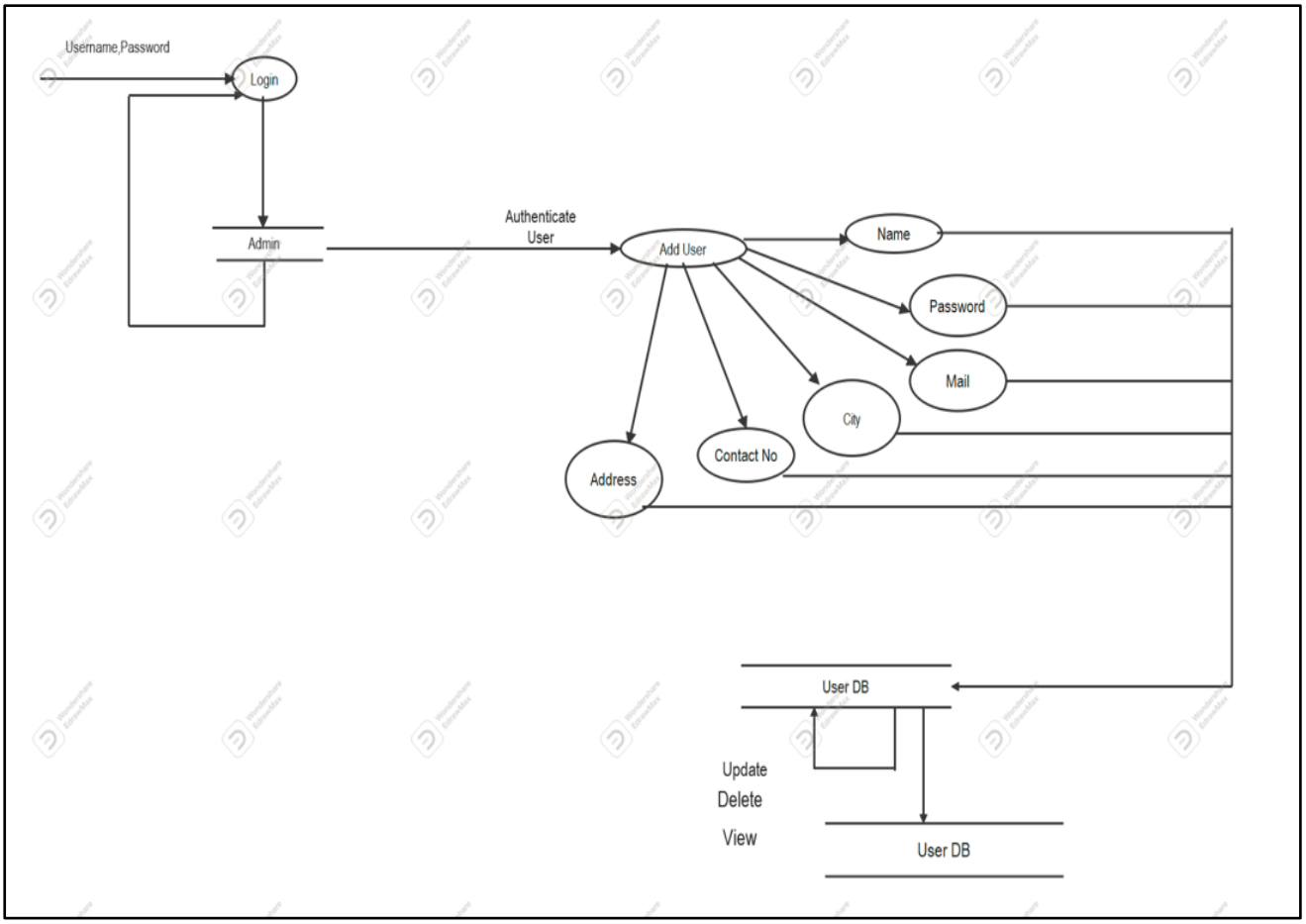
DFD (DATA FLOW DIAGRAM):ZERO LEVEL DFD



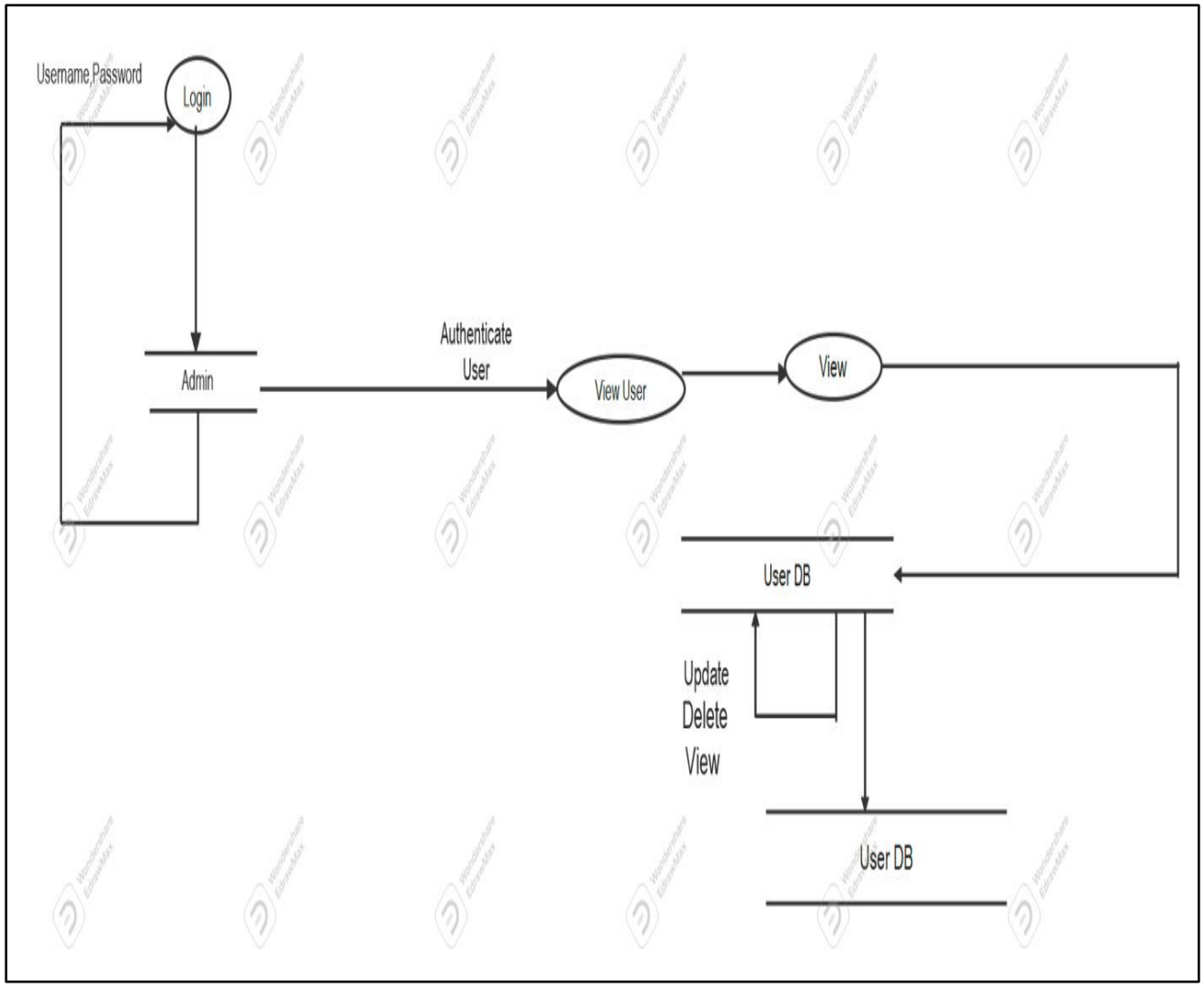
DFD (DATA FLOW DIAGRAM): LEVEL ONE DFD



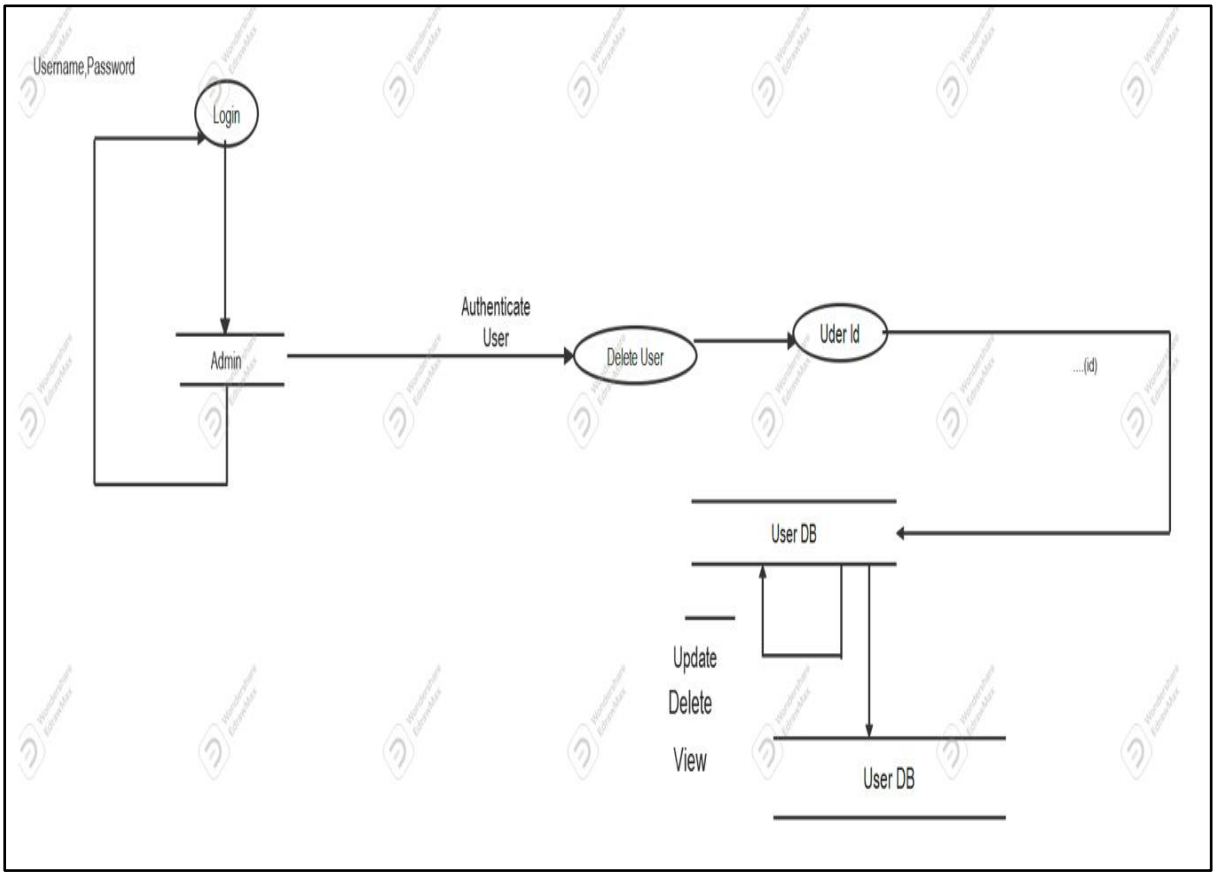
DFD (DATA FLOW DIAGRAM):SECOND LEVEL DFD(2.A1)



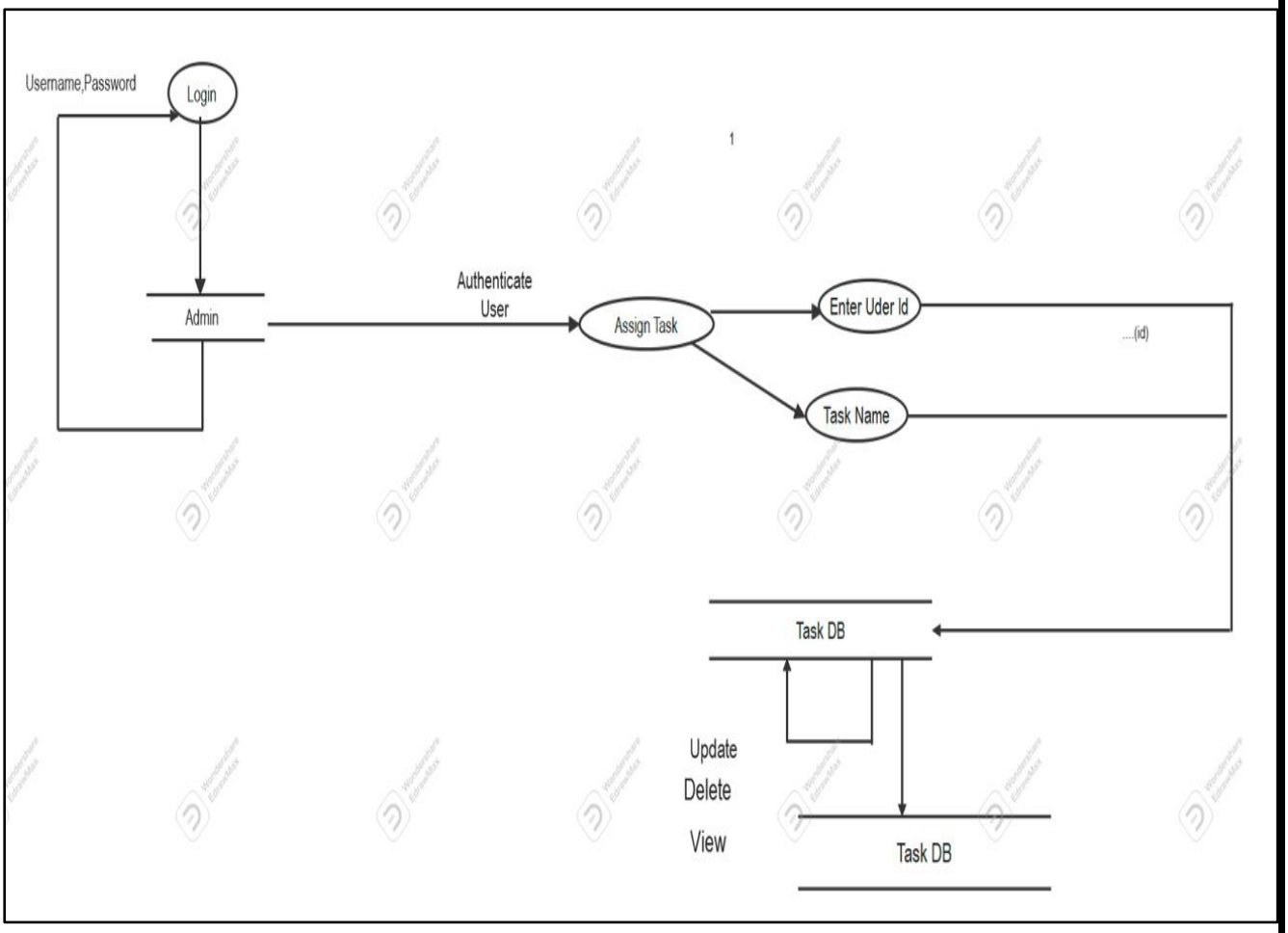
DFD(DATA FLOW DIAGRAM):SECOND LEVEL DFD(2.A2)



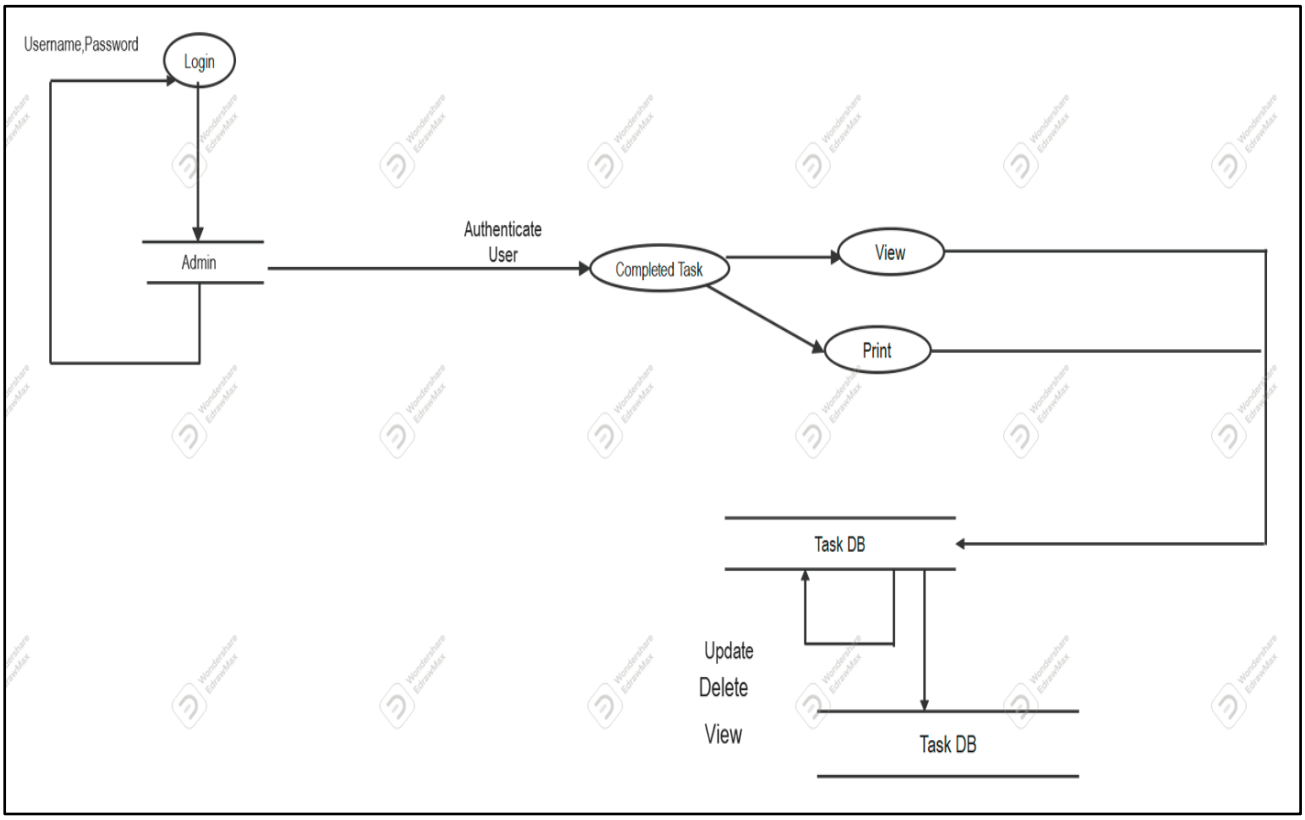
DFD(DATA FLOW DIAGRAM):SECOND LEVEL DFD(2.A3)



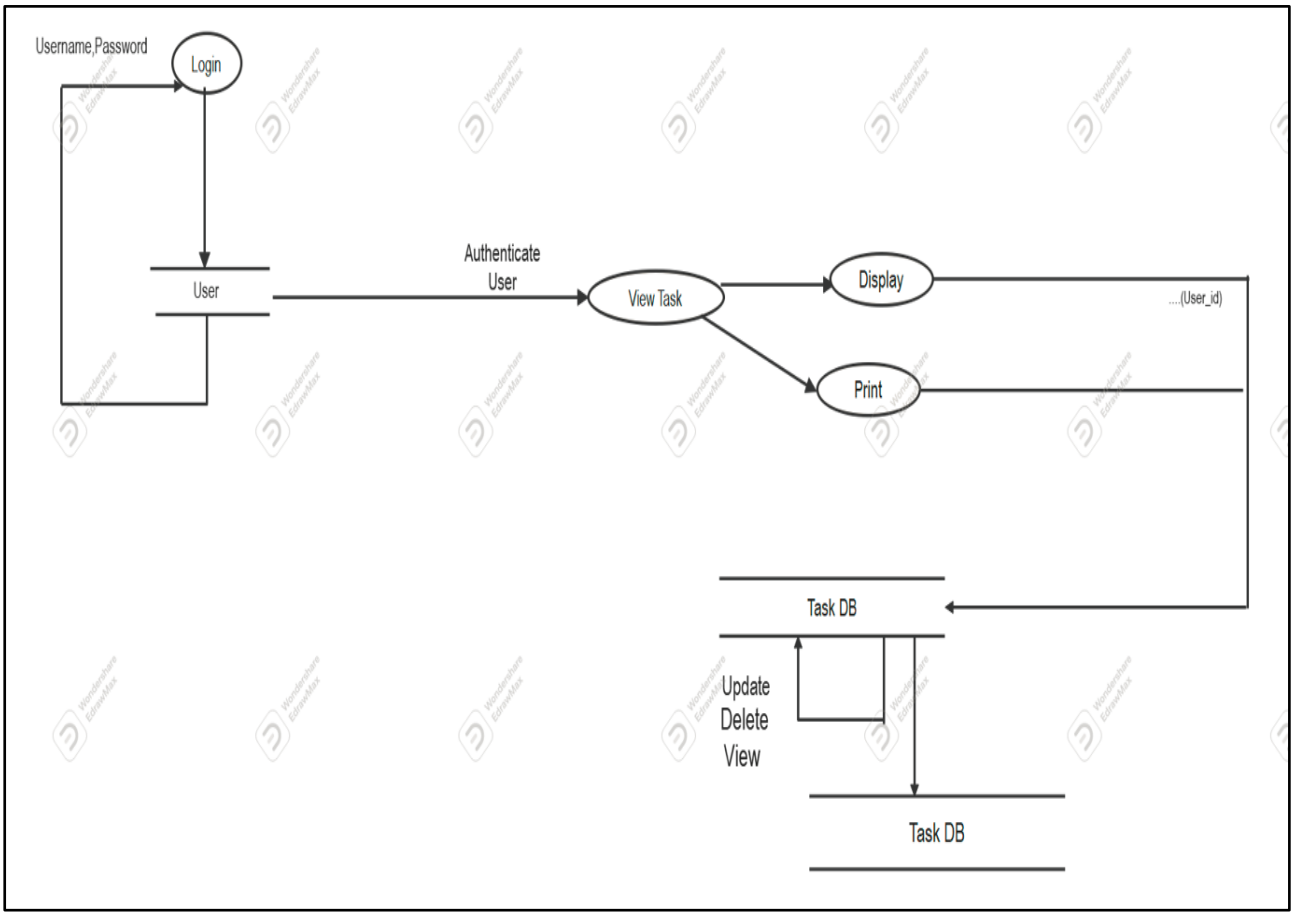
DFD(DATA FLOW DIAGRAM):SECOND LEVEL DFD(2.A4)



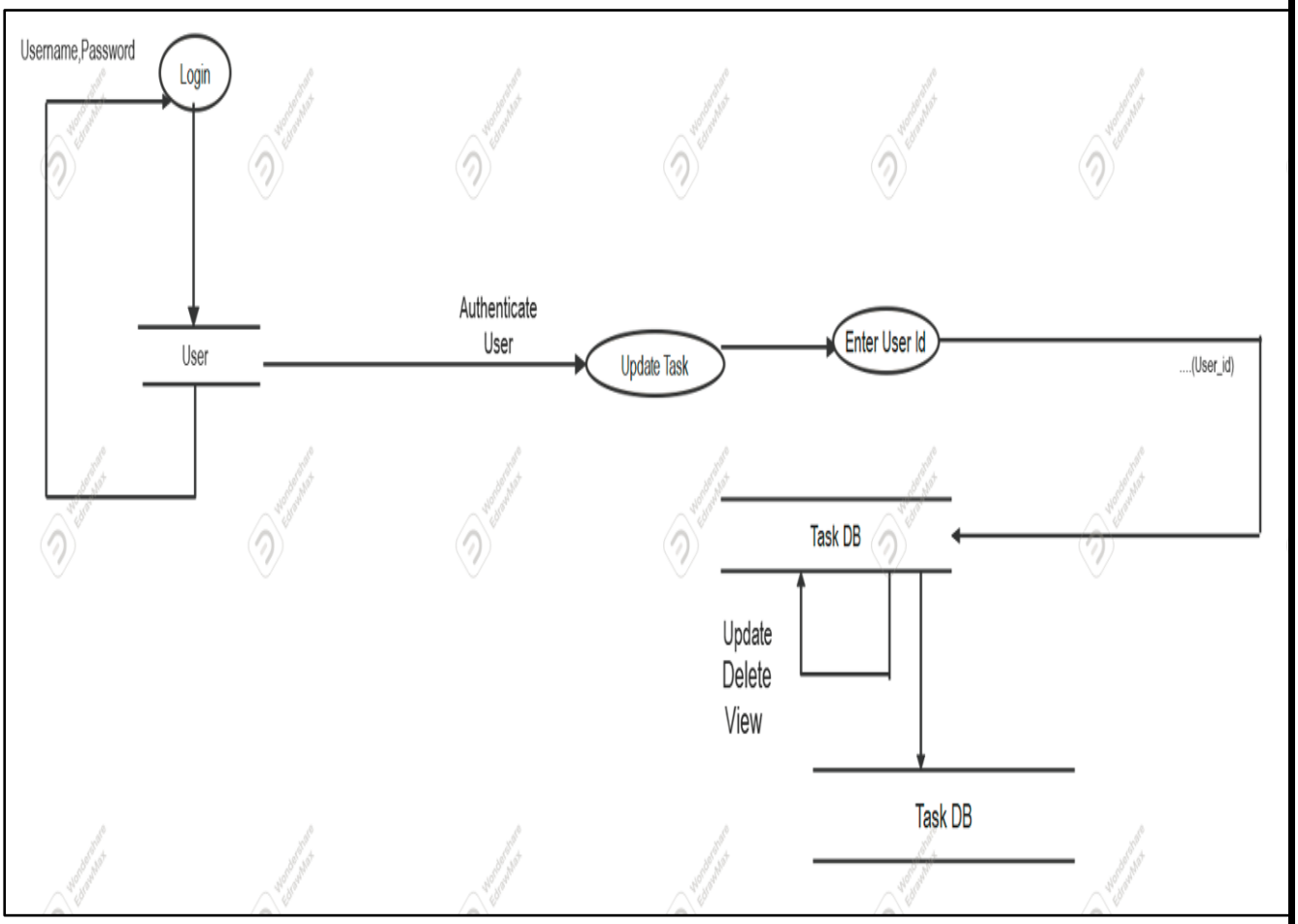
DFD(DATA FLOW DIAGRAM):SECOND LEVEL DFD(2.A5)



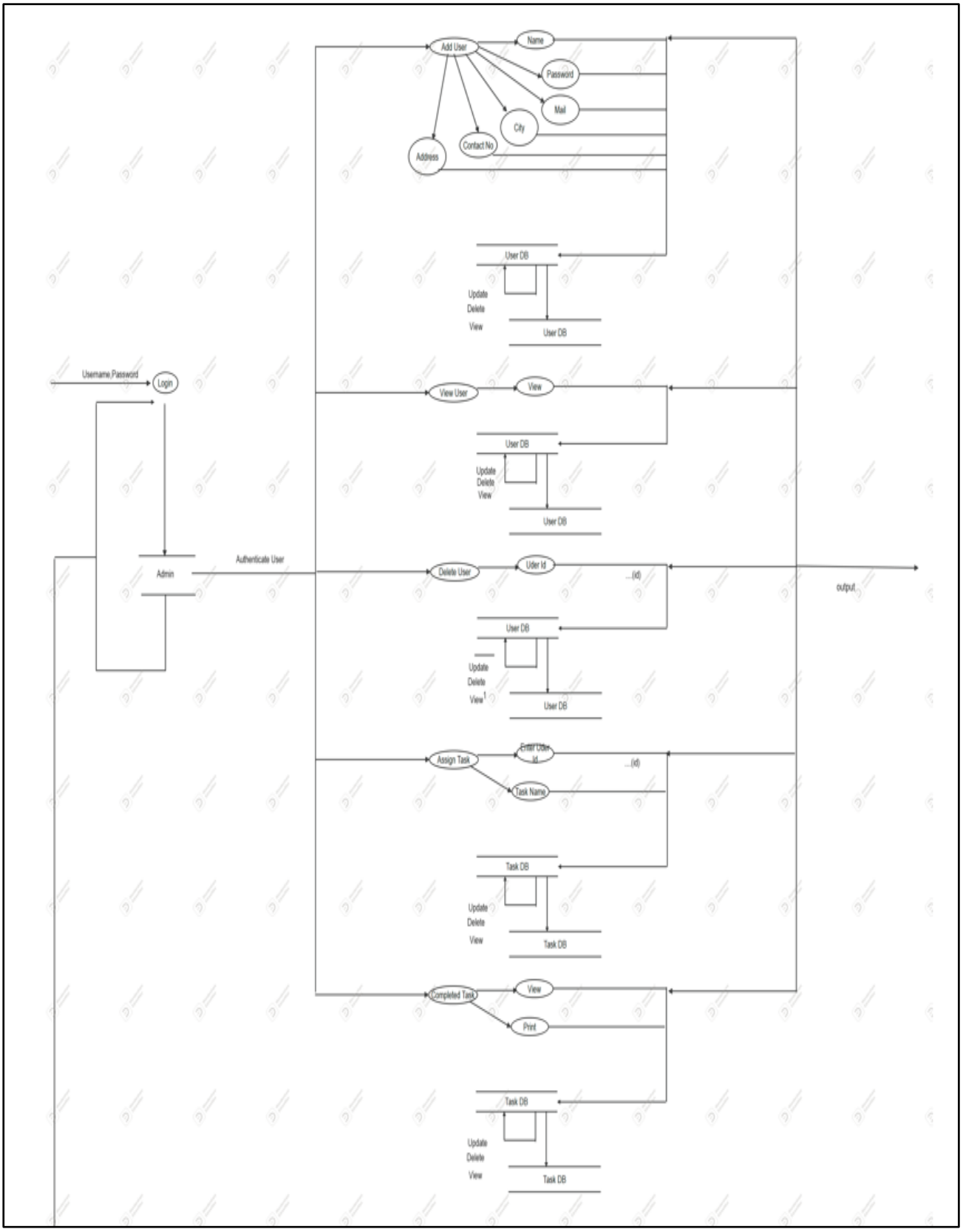
DFD(DATA FLOW DIAGRAM):<u>SECOND LEVEL DFD(2.B1)</u>

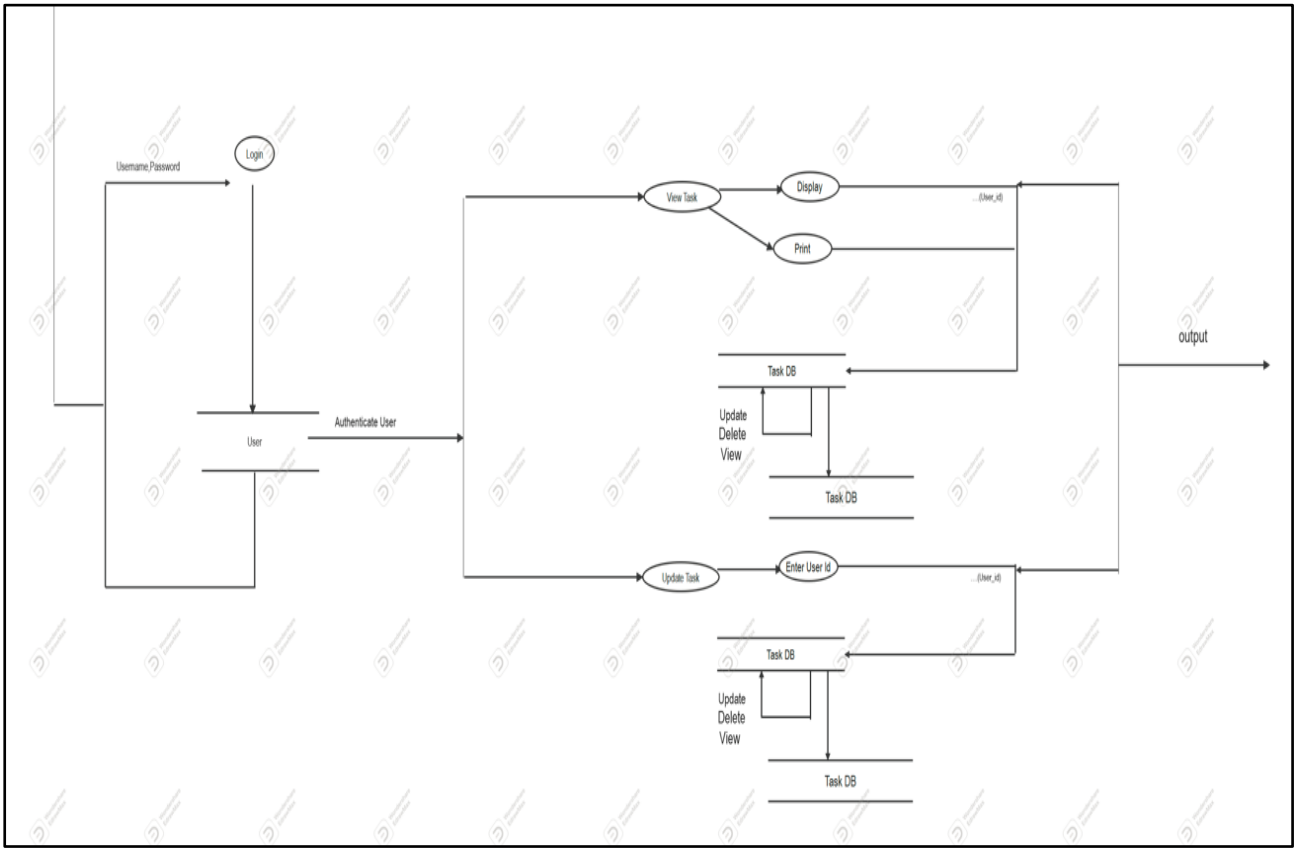


DFD(DATA FLOW DIAGRAM):<u>SECOND LEVEL DFD(2.B2)</u>

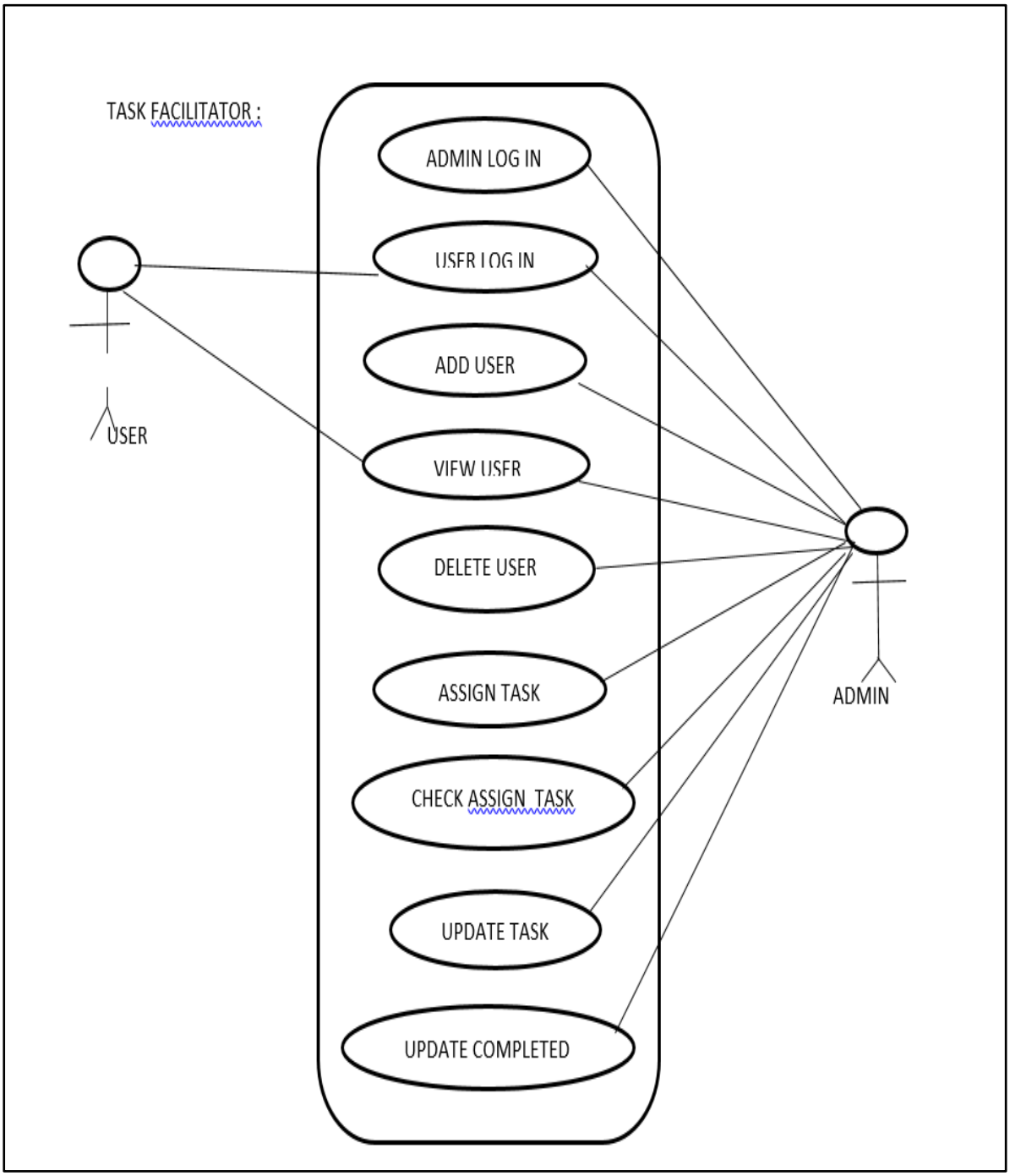


DFD(DATA FLOW DIAGRAM):THIRD LEVEL DFD

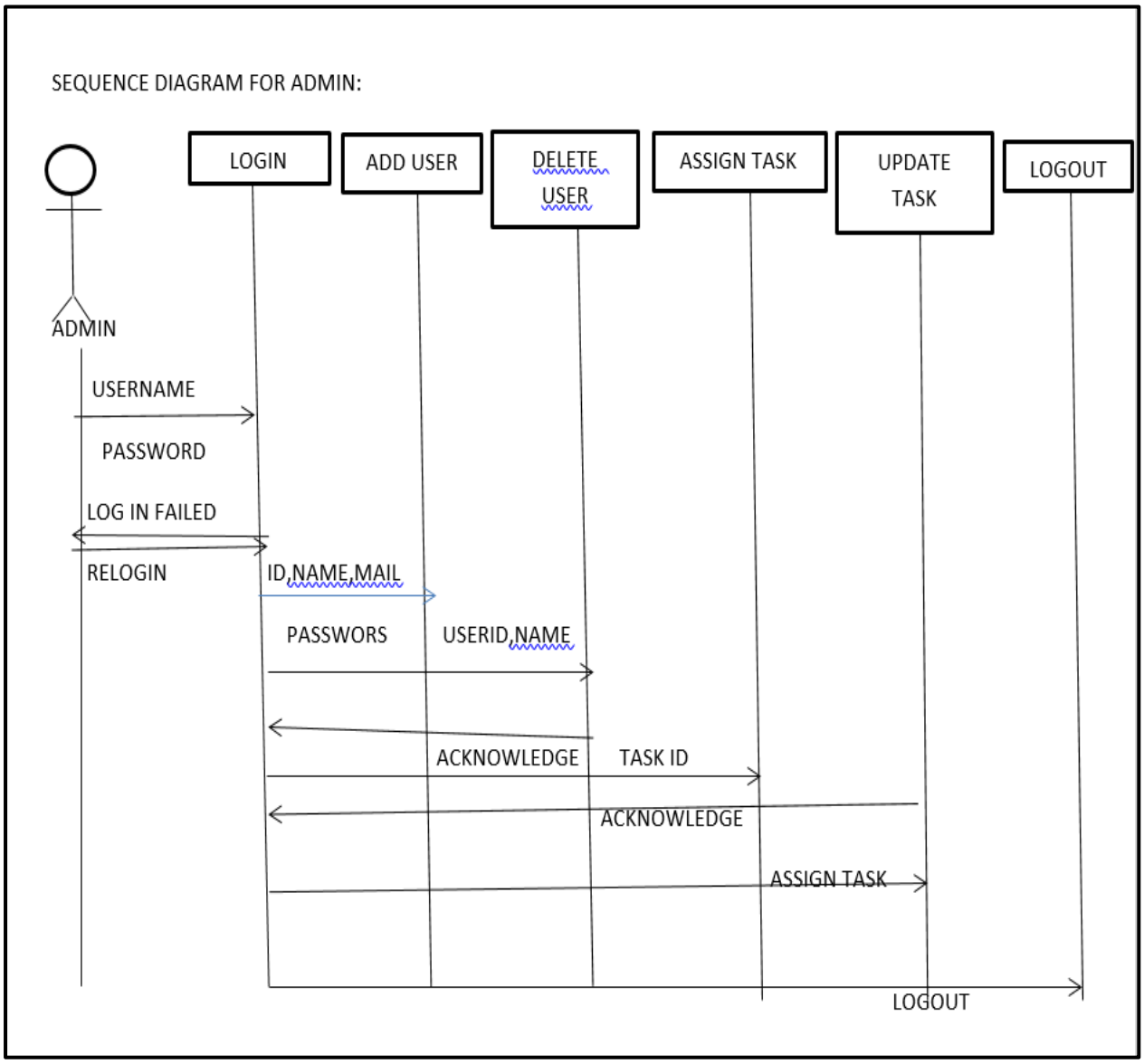



Login
Username,Password
User
Authenticate User
View Task
Display
Print
Task DB
Update
Delete
View
Task DB
output
Update Task
Enter User Id
Task DB
Update
Delete
View
Task DB

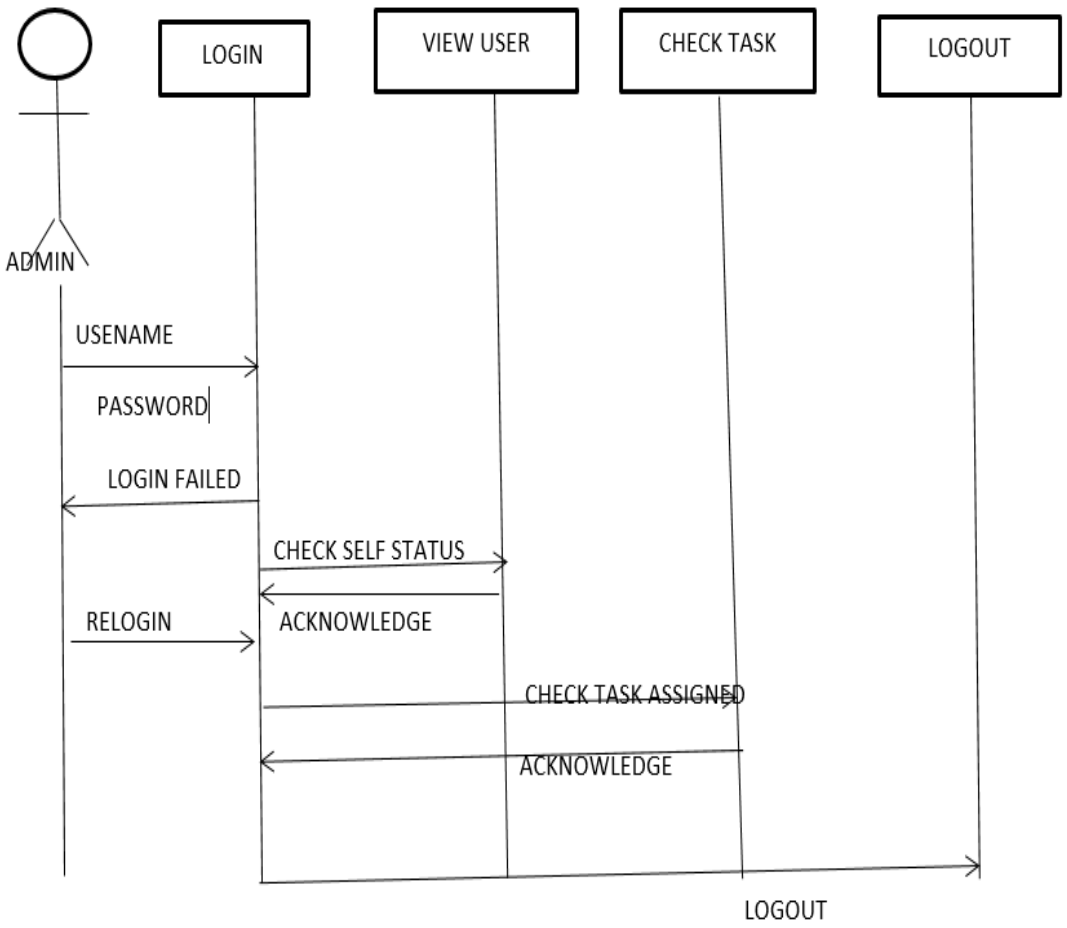
3.3) USE CASE DIAGRAM:



3.4) SEQUENCE DIAGRAM:

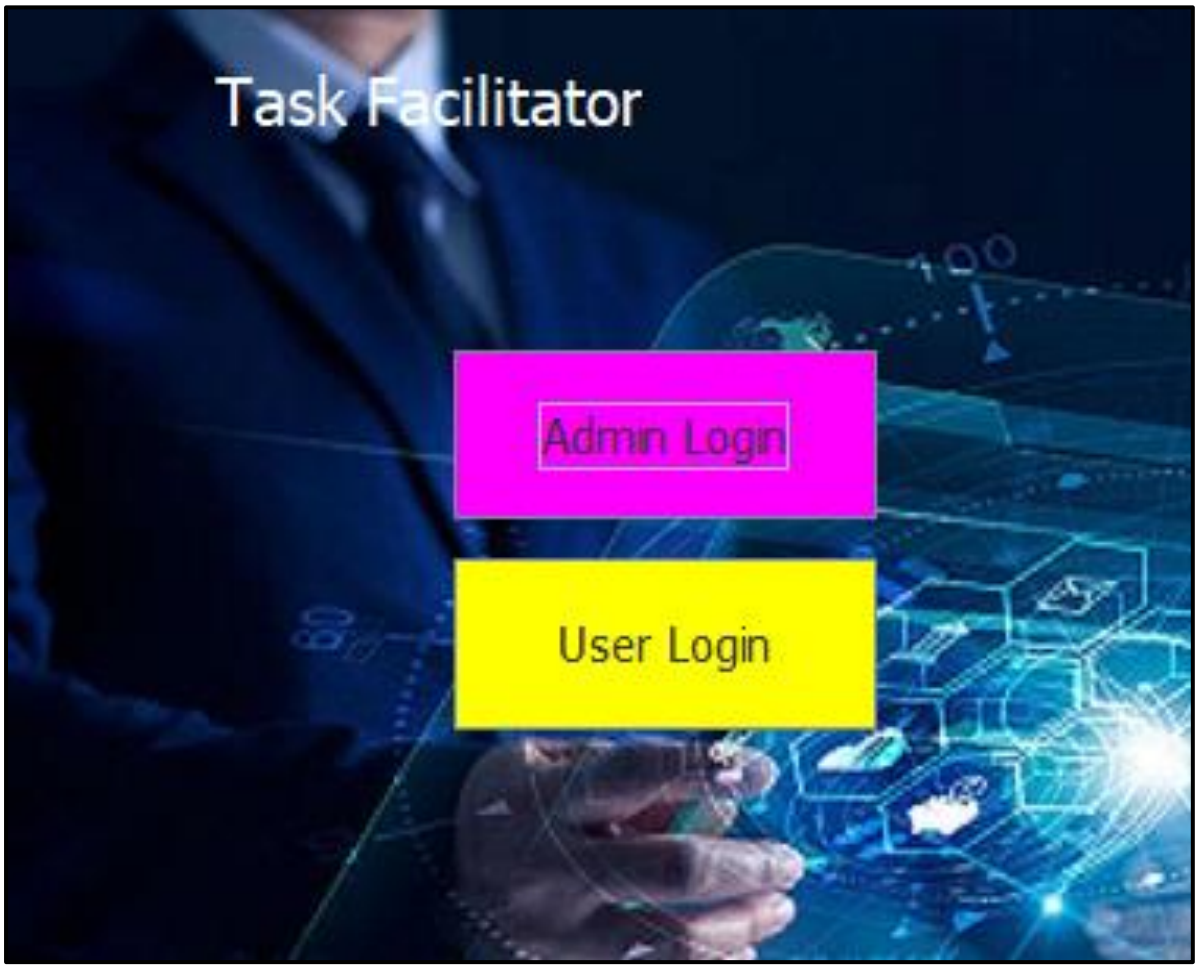


SEQUENCE DIAGRAM FOR USER:



3.5) USER INTERFACE DESIGN(SCREENS):

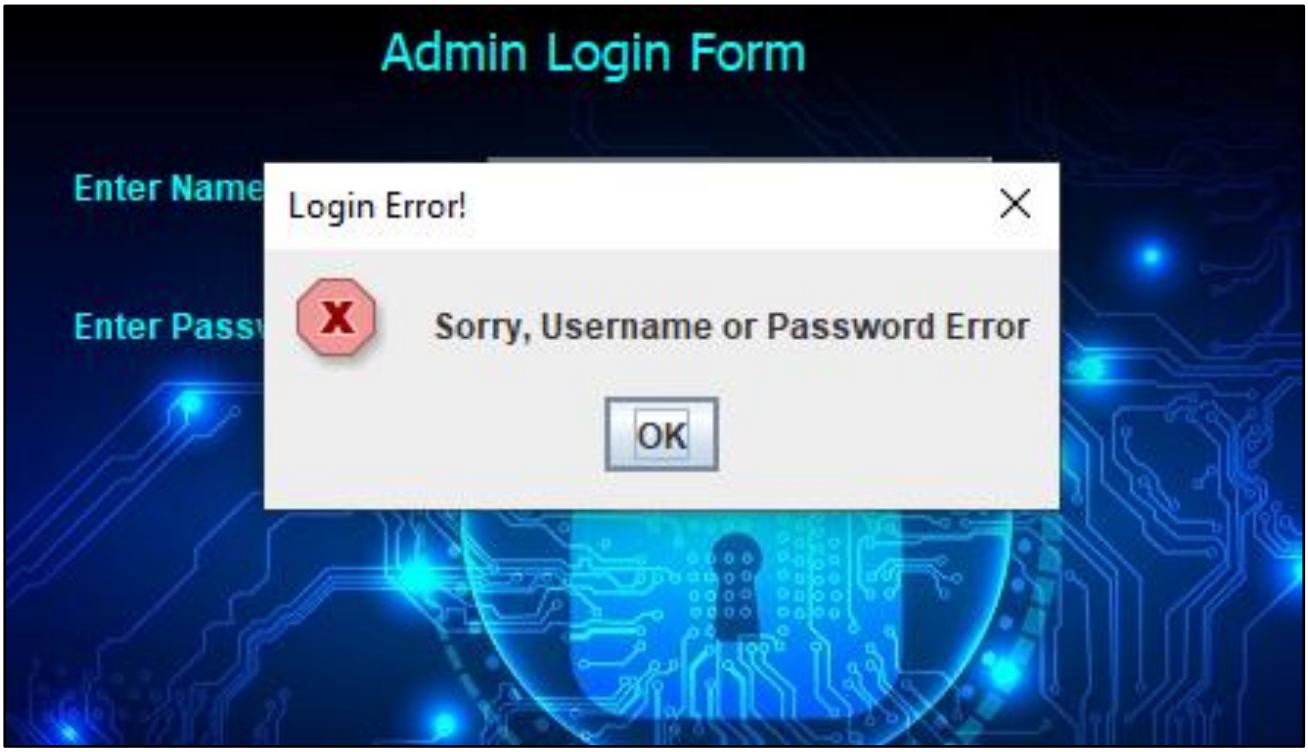
- **ADMIN LOGIN:**



- **ADMIN LOGIN FORM:**

- **VALIDATION FOR USERNAME AND PASSWORD:**

IF USER PUT WRONG USER NAME AND PASSWORD ITS SHOW'S THE ERROR TOP UP DIALOG BOX.



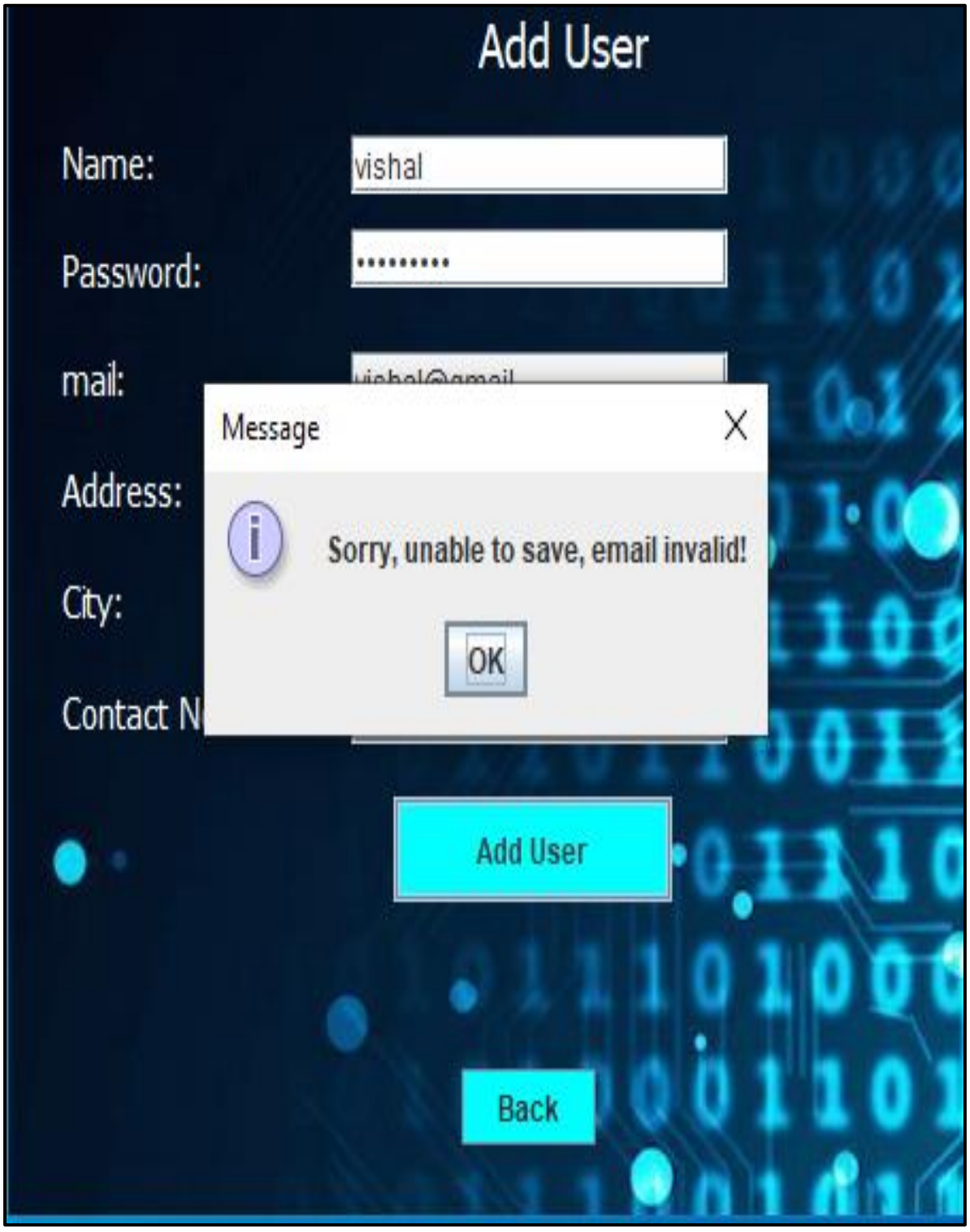
- **ADMIN SECTION:**

- **ADD USER :**

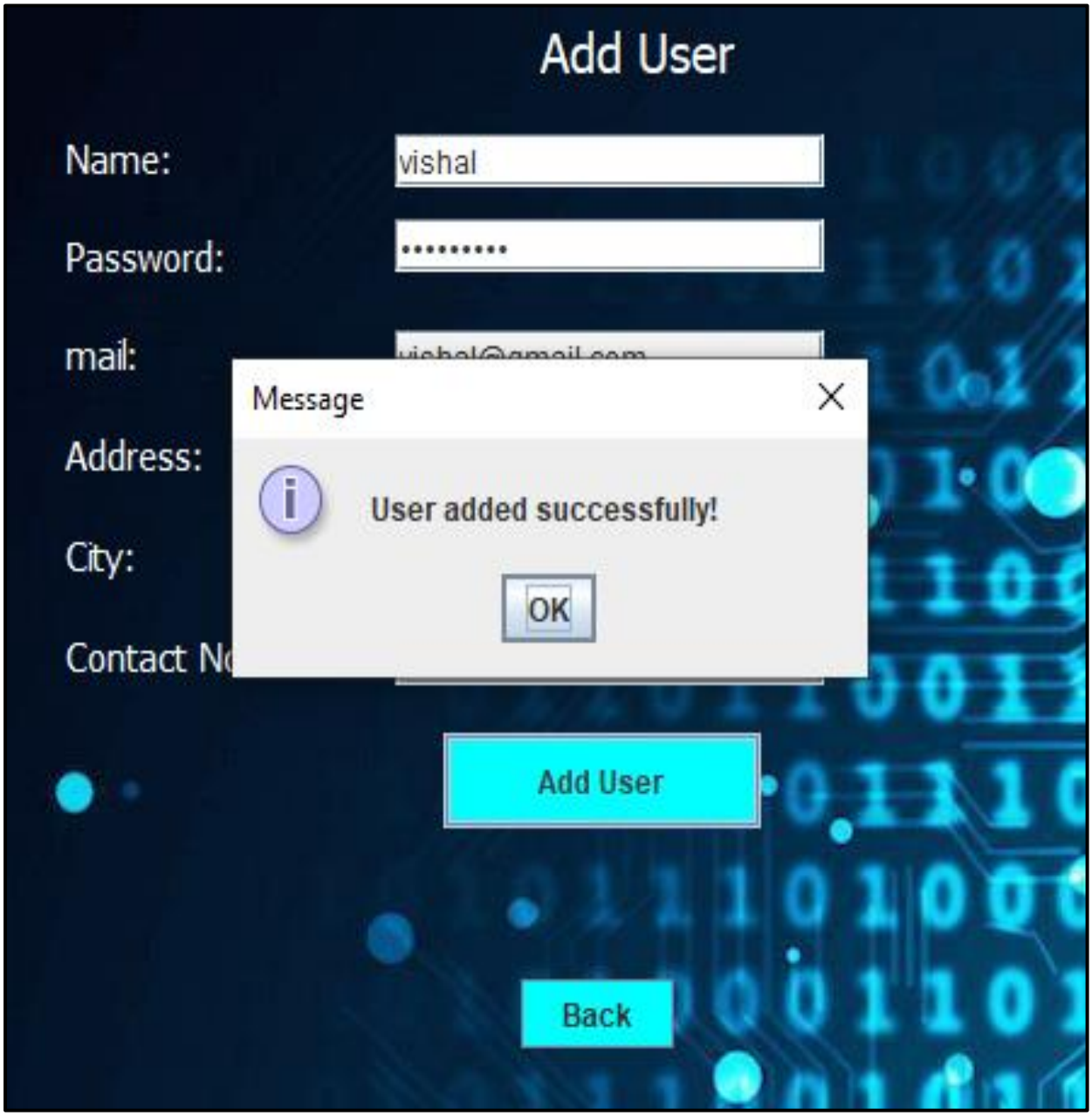

Add User
Name:
Password:
mail:
Address:
City:
Contact No:
Add User
Back


Add User
Name:
vishal
Password:
mail:
vishal@gmail.com
Address:
tmc rd
City:
thane
Contact No:
9874566541
Add User
Back

- **VALIDATION FOR EMAIL:**



WHEN USER ADDED SUCCESFULLY IT SHOW THE DIALOG BOX..THAT USER ADDED SUCCESFULLY.

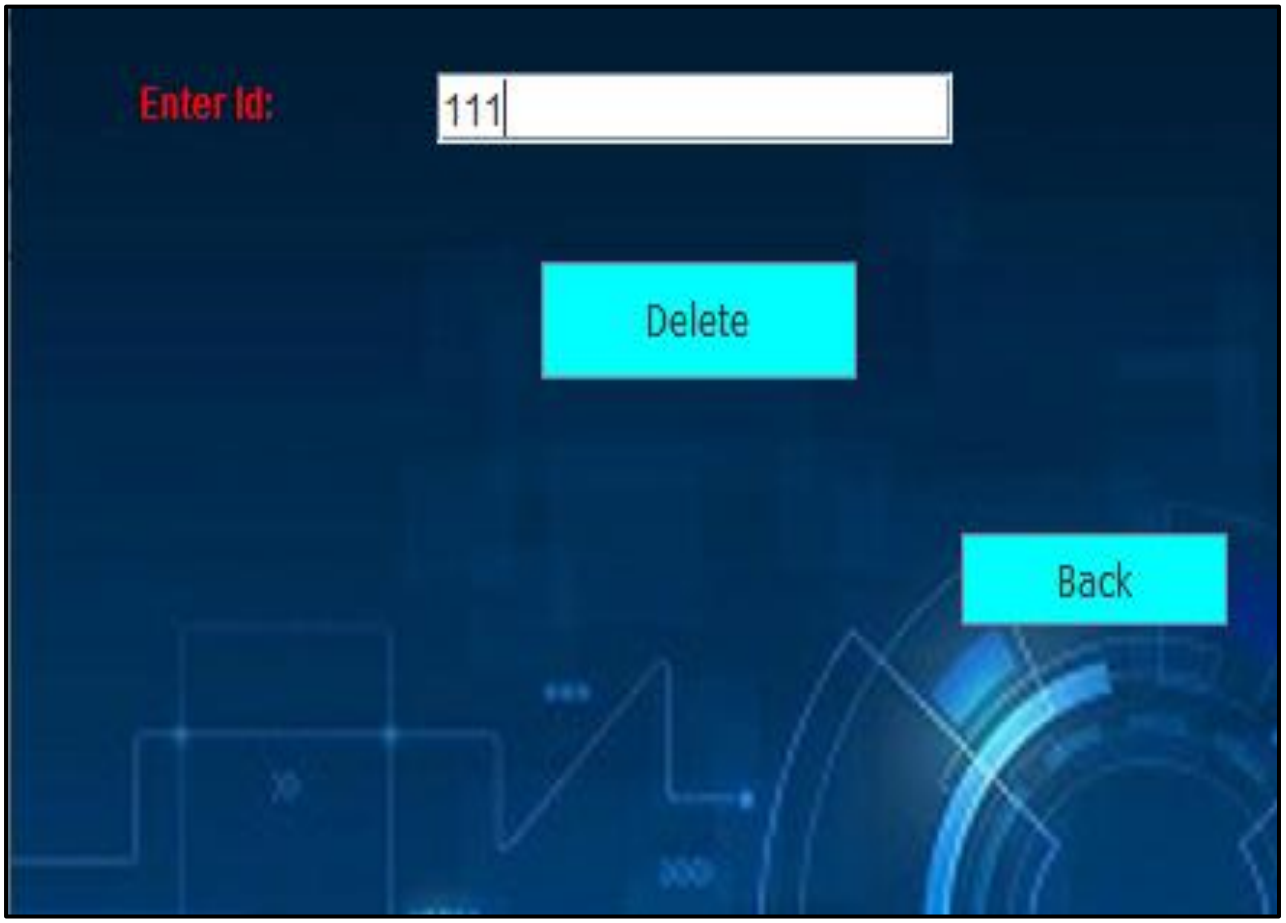


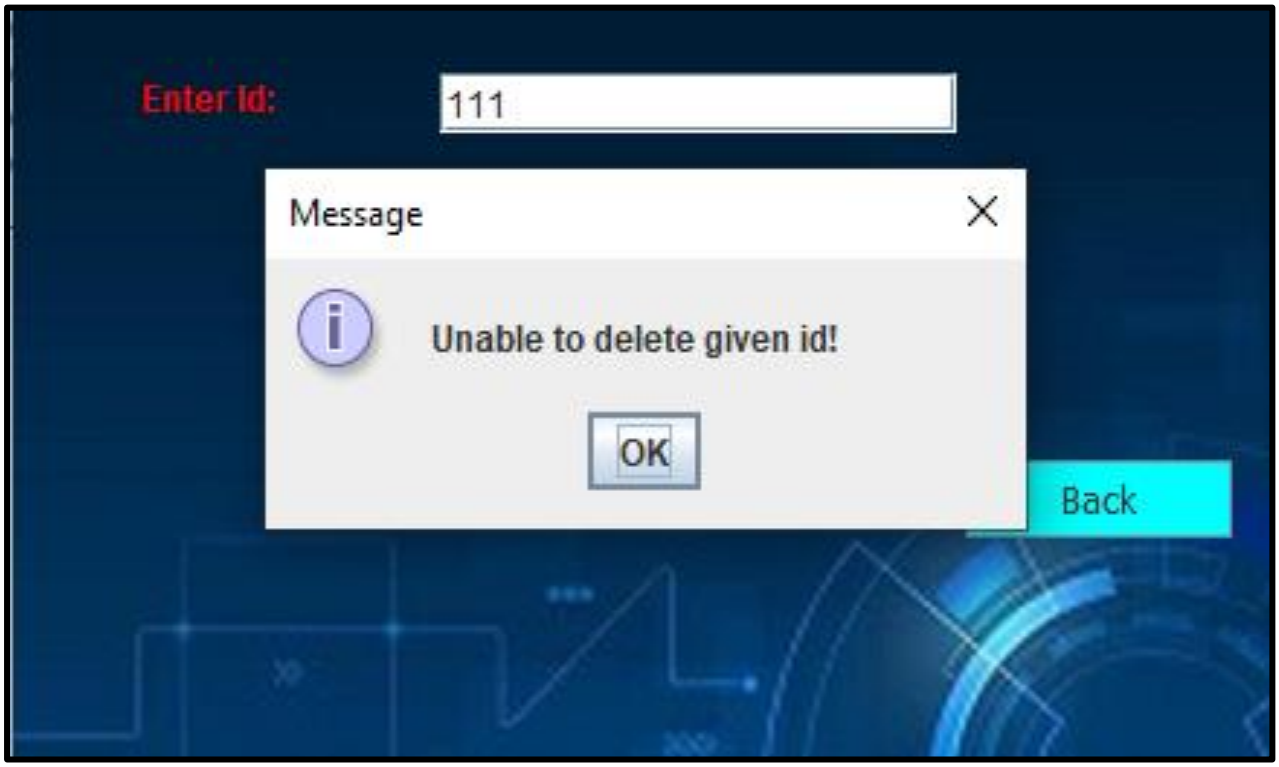
- **VIEW USER :**

id	name	passw...	email	address	city	contact
1	abc	abc	abc@g...	my-tm...	pune	12345...
2	akshay	akshay...	akshay...	khar	mumbai	77896...
3	sakshi	sakshi...	sakshi...	pcmc	pune	7963264
4	sameer	samee...	sam@...	kpg	ahema...	77684...
5	ajay	ajay123	ajay@g...	pc	pune	87965...
7	smit	smit123	smit@...	gb	thane	78965...
8	nargis	nargis...	nargis...	pc	pune	96784...
11	suraj	suraj123	suraj@...	pc	pune	96197...
14	ashuto...	ashu123	ashu@...	abc	xyz	77854...
15	sachin	sachin...	sachin...	chembur	mumbai	74813...
16	rohan	ron123	ron@g...	bcabin	thane	77302...
17	vishal	vishal1...	vishal...	tmc rd	thane	98745...

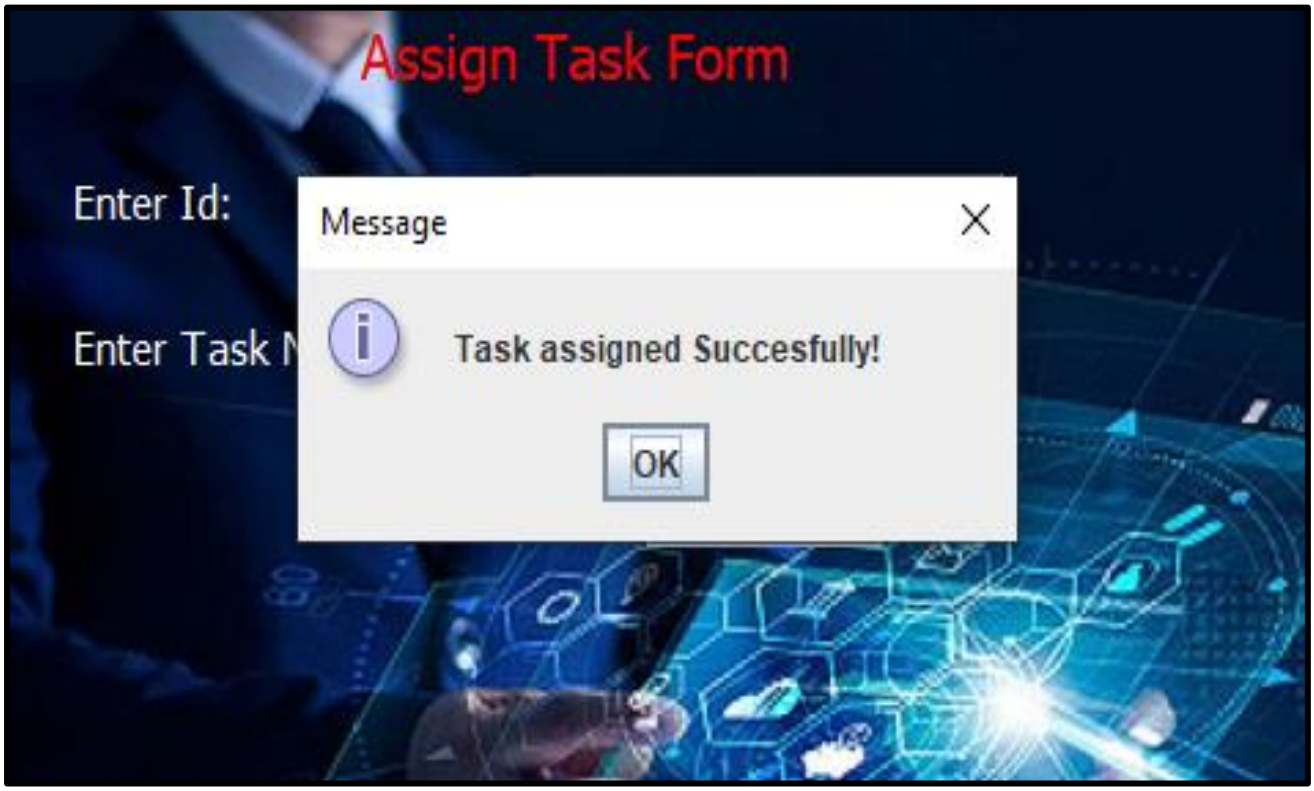
Back

- **DELETE USER :**

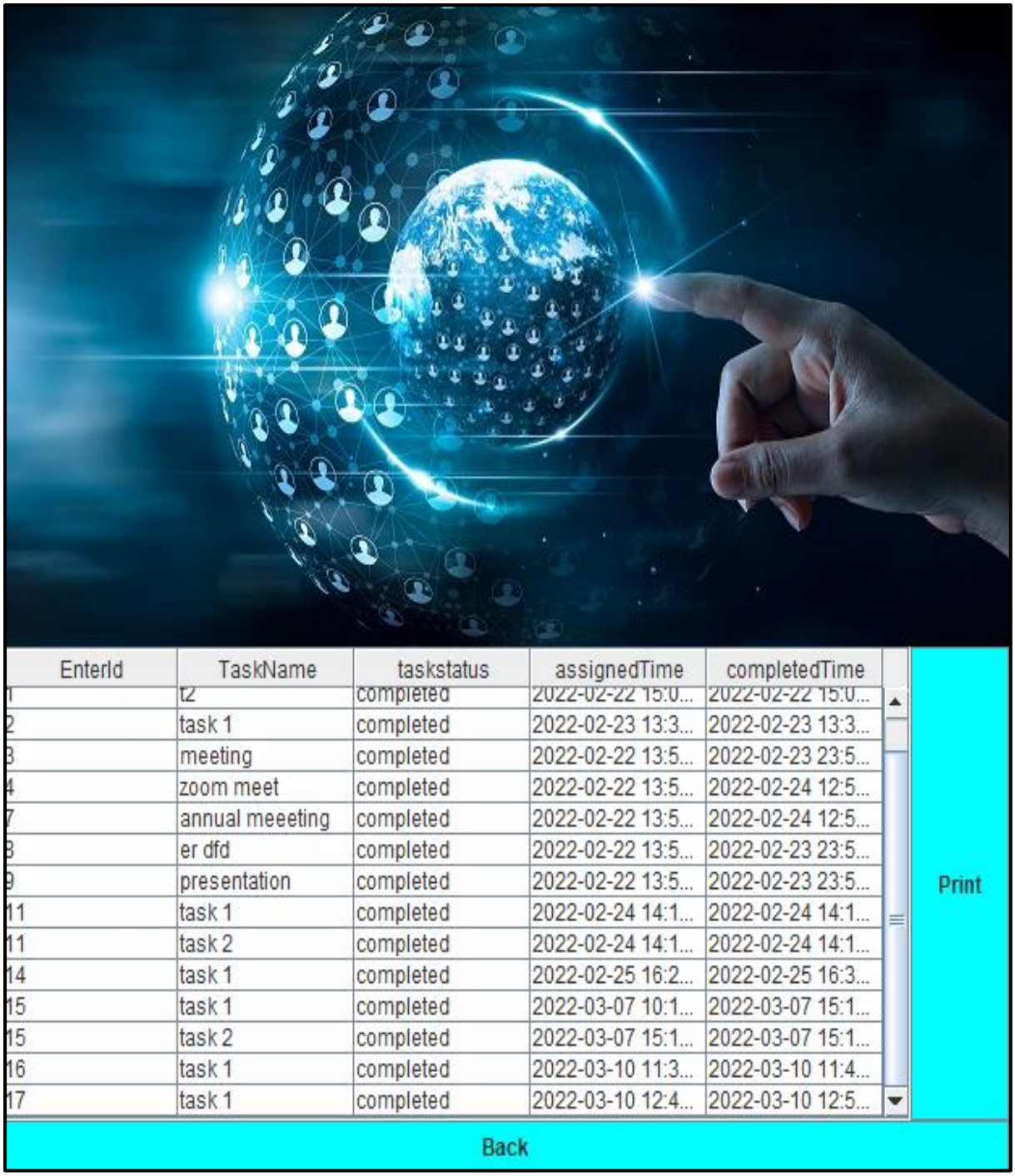

Enter Id:
111
Delete
Back


Enter Id:
111
Message
Unable to delete given id!
OK
Back

- **ASSIGN TASK TO USER :**


Assign Task Form
Enter Id:
Enter Task
Message
Task assigned Succesfully!
OK

- **COMPLETEED TASK FROM USER :**

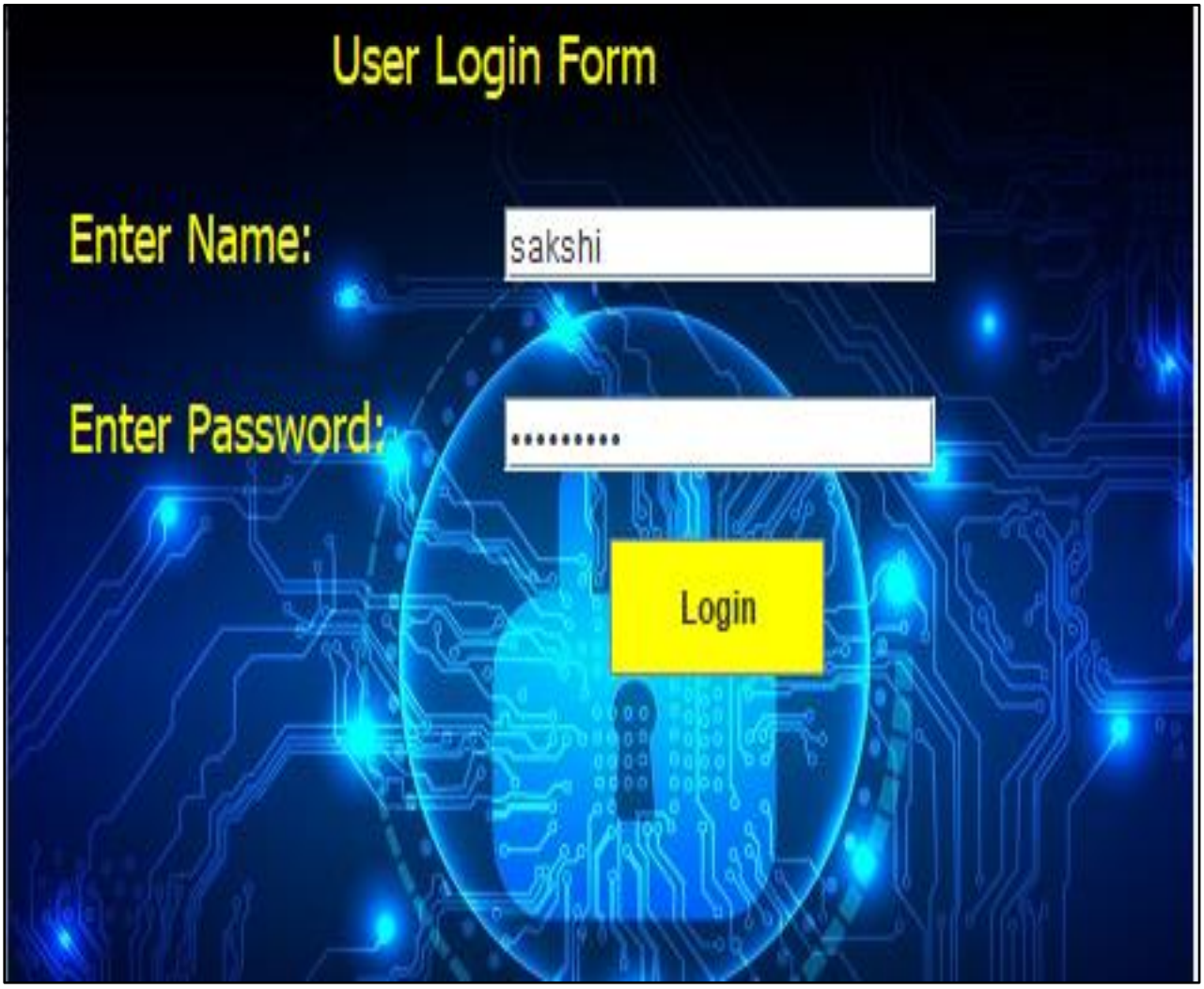


EnterId	TaskName	taskstatus	assignedTime	completedTime
1	t2	completed	2022-02-22 15:0...	2022-02-22 15:0...
2	task 1	completed	2022-02-23 13:3...	2022-02-23 13:3...
3	meeting	completed	2022-02-22 13:5...	2022-02-23 23:5...
4	zoom meet	completed	2022-02-22 13:5...	2022-02-24 12:5...
7	annual meeeting	completed	2022-02-22 13:5...	2022-02-24 12:5...
8	er dfd	completed	2022-02-22 13:5...	2022-02-23 23:5...
9	presentation	completed	2022-02-22 13:5...	2022-02-23 23:5...
11	task 1	completed	2022-02-24 14:1...	2022-02-24 14:1...
11	task 2	completed	2022-02-24 14:1...	2022-02-24 14:1...
14	task 1	completed	2022-02-25 16:2...	2022-02-25 16:3...
15	task 1	completed	2022-03-07 10:1...	2022-03-07 15:1...
15	task 2	completed	2022-03-07 15:1...	2022-03-07 15:1...
16	task 1	completed	2022-03-10 11:3...	2022-03-10 11:4...
17	task 1	completed	2022-03-10 12:4...	2022-03-10 12:5...

Print

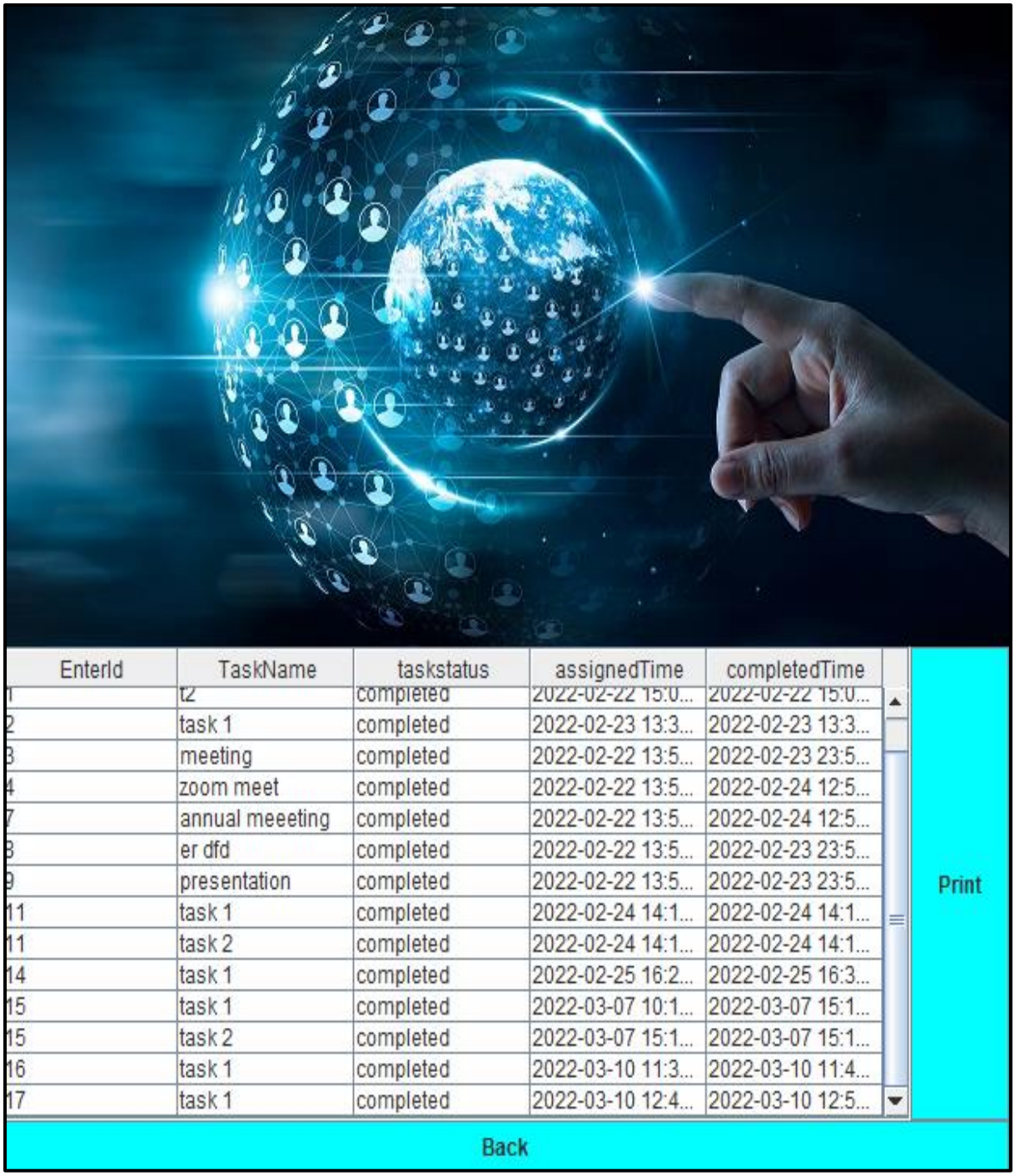
Back

- **USER LOGIN:**



- **USER SECTION :**

- **VIEW TASK :**

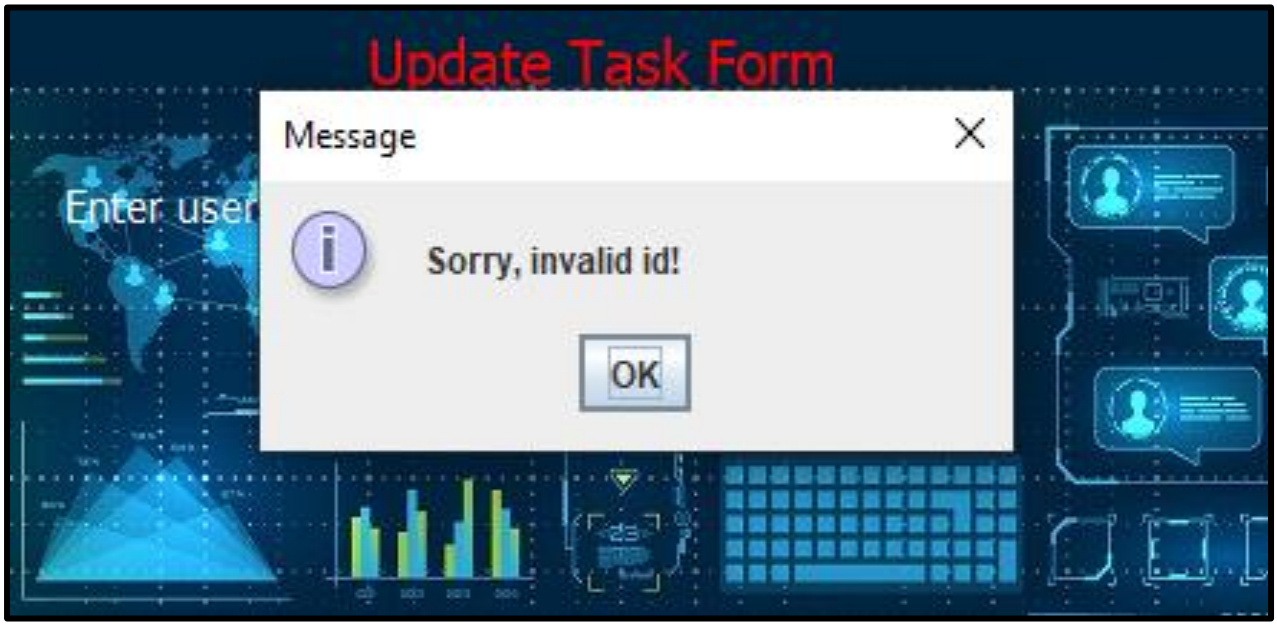


EnterId	TaskName	taskstatus	assignedTime	completedTime
1	t2	completed	2022-02-22 15:0...	2022-02-22 15:0...
2	task 1	completed	2022-02-23 13:3...	2022-02-23 13:3...
3	meeting	completed	2022-02-22 13:5...	2022-02-23 23:5...
4	zoom meet	completed	2022-02-22 13:5...	2022-02-24 12:5...
7	annual meeeting	completed	2022-02-22 13:5...	2022-02-24 12:5...
8	er dfd	completed	2022-02-22 13:5...	2022-02-23 23:5...
9	presentation	completed	2022-02-22 13:5...	2022-02-23 23:5...
11	task 1	completed	2022-02-24 14:1...	2022-02-24 14:1...
11	task 2	completed	2022-02-24 14:1...	2022-02-24 14:1...
14	task 1	completed	2022-02-25 16:2...	2022-02-25 16:3...
15	task 1	completed	2022-03-07 10:1...	2022-03-07 15:1...
15	task 2	completed	2022-03-07 15:1...	2022-03-07 15:1...
16	task 1	completed	2022-03-10 11:3...	2022-03-10 11:4...
17	task 1	completed	2022-03-10 12:4...	2022-03-10 12:5...

Print

Back

- **UPDATE TASK :**


Update Task Form
Message
Enter user
Sorry, invalid id!
OK

3.6) DATA DICTIONARY:

- **TABLE 1:-USER**

Field Name	DataType	Size	Constraints
id	Integer	NULL	Primary key
Name	Varchar	255	Not Null
Password	Varchar	255	Not Null
Email	Varchar	255	Not Null
Address	Varchar	255	Not Null
City	Varchar	255	Not Null
Contact	Varchar	255	Not Null

- **TABLE 2:TASK**

Field Name	DataType	Size	Constraints
id	Int	NULL	Primary key
Task_Name	Varchar	255	Not Null
Task_Status	Varchar	30	Not Null

3.7)TEST PROCEDURE AND IMPLEMENTATION

- The software testing is the critical element of software quality assurance and represents the ultimate review of the software design and coding. The main objective of the testing is to find an error and to uncover the errors that are not yet discovered.
- The increasing visibility of software as a system element and the attendant cost associated with a software failure and motivating forces for well planned, through testing. It is no unusual for a software development organization to expand between 30% to 40% of project effort on testing. In the extreme, testing of human related software can cost 3-5 time as much as all other software engineering activities combined. the testing phase involves the testing of the system using various test data, preparation of the test data plays a vital role in the system testing after preparing the test data, error where found and corrected by using the following the testing steps and correction are recorded for future reference. Thus a series of testing is performed on the system before it is ready for implementation.
- After completion of system analysis, design and coding through testing of the system was carried out in a systematic approach, the main objectives of the system are

➢ To ensure that the operations of the system will perform as per the specification.

➢ To make sure that the system meets the user requirement during the operations.

➢ To cross check the when correct input are filled into the system output are correct.

➢ To make sure that during the operation incorrect inputs and the outputs will be detected.

In testing process the number of strategies have been used as mentioned below, • Unit Testing

• Integration Testing

• Validation Testing

• Black Box Testing

• User acceptance Testing Unit Testing

Unit testing focuses verification efforts on the smallest unit of the software design.

Using the system test plan, prepare in the design phase of the system development as guide, important control path are tested to uncover error within boundary of the module. The interface of each of the module was tested to ensure proper flow of information into and out of the module under consideration. Each module will be tested individually so as to make the individual component error free. Also other attached modules will be error free.

Integration Testing:

Each module will be tested of its effect on other module by integrating the modules.

This will remove further errors from the system and may also result in some changes in the

individual module.

Validation Testing

At the culmination of the integration testing the software was completely assembled as package, interfaces have been uncovered, and a final series of software validation testing began. Here we test the system function manner that can be reasonably by the users ,the system was tested against system requirement specification.

- Black Box Testing:

After performing validation testing, the next phase is output test of the system, since no system code is useful if it does not produce the desired output in desired format. By considering the format of the report/output, report/output is generated or displayed and tested.

- User Acceptance Testing:

User acceptance testing is used to determine the whether the software is fit for the user to use. The System under consideration was listed for user acceptance by keeping constant touch with the prospective user of the system at the time of design, development and making change whenever required.

CHAPTER 4 : USER MANUAL

4.1) USER MANUAL

This manual contains information how to operate Task Facilitator system in application where the user can handle basic functions.

A Admin has access to functionalities of adding User and view User, Assign Task and Delete User .

1- Admin :

Admin will have the full authority of the software.

Admin will login by using his account.

Admin will Add , view and Delete the details of user.

2- Users :

User will have the authority of View task.

User can Update the task status.

4.2) OPERATIONAL MANUAL/ MENU MANUAL

1- Login

Login is the very first form when you start debugging. Admin required to login by usernsme and password. The system allows access to system if they are valid.

2- Add User Form

Using this form Admin can add new User in system. System validates all required fields and insert a user details record if they are valid.

3- Assign Form

Using this form Admin can assign task to user .

4- Update Task Form

Using this form user can update the status of task assign .

5- View and Delete Form

Using this form Admin can view details of user and delete the user form System.

4.3) REPORTS

Generate Report

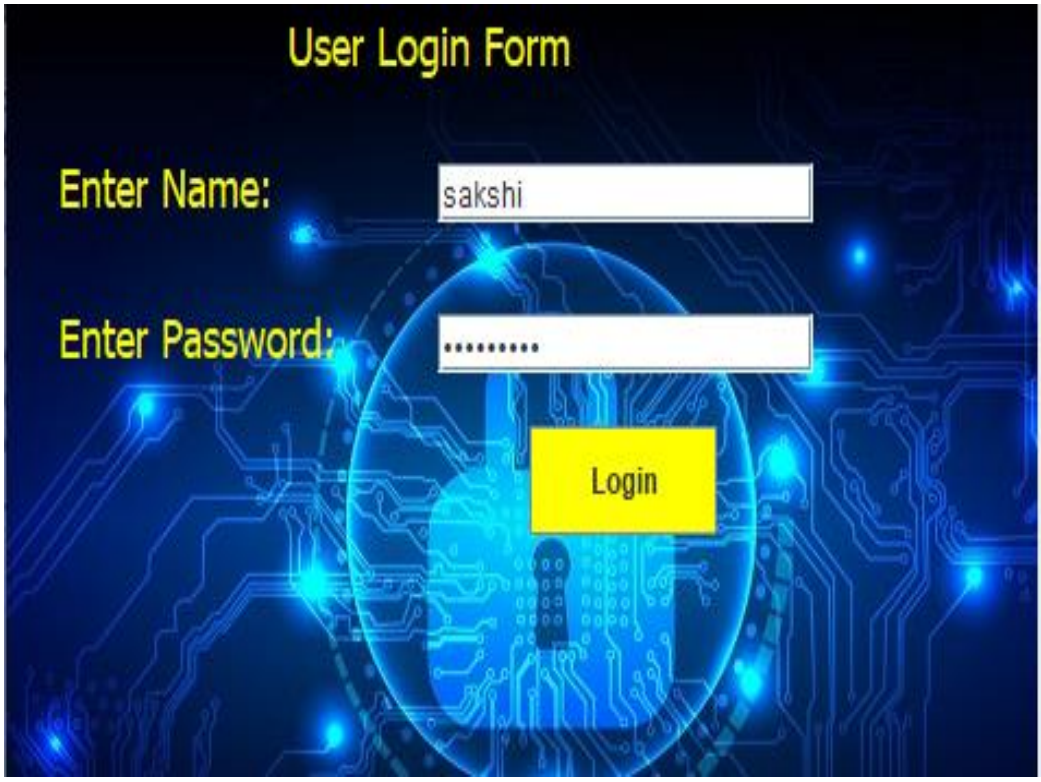
The REPORT contains information about User details, Task details, Task Status details etc.

CHAPTER 5 – ANNEXURES

5.1) USER INTERFACE SCREEN

- **ADMIN LOGIN:**

- **USER LOGIN:**



- **USER SECTION:**

➢ VIEW TASK:

EnterId	TaskName	taskstatus	assignedTime	completedTime
1	t2	completed	2022-02-22 15:0...	2022-02-22 15:0...
2	task 1	completed	2022-02-23 13:3...	2022-02-23 13:3...
3	meeting	completed	2022-02-22 13:5...	2022-02-23 23:5...
4	zoom meet	completed	2022-02-22 13:5...	2022-02-24 12:5...
7	annual meeeting	completed	2022-02-22 13:5...	2022-02-24 12:5...
8	er dfd	completed	2022-02-22 13:5...	2022-02-23 23:5...
9	presentation	completed	2022-02-22 13:5...	2022-02-23 23:5...
11	task 1	completed	2022-02-24 14:1...	2022-02-24 14:1...
11	task 2	completed	2022-02-24 14:1...	2022-02-24 14:1...
14	task 1	completed	2022-02-25 16:2...	2022-02-25 16:3...
15	task 1	completed	2022-03-07 10:1...	2022-03-07 15:1...
15	task 2	completed	2022-03-07 15:1...	2022-03-07 15:1...
16	task 1	completed	2022-03-10 11:3...	2022-03-10 11:4...
17	task 1	completed	2022-03-10 12:4...	2022-03-10 12:5...

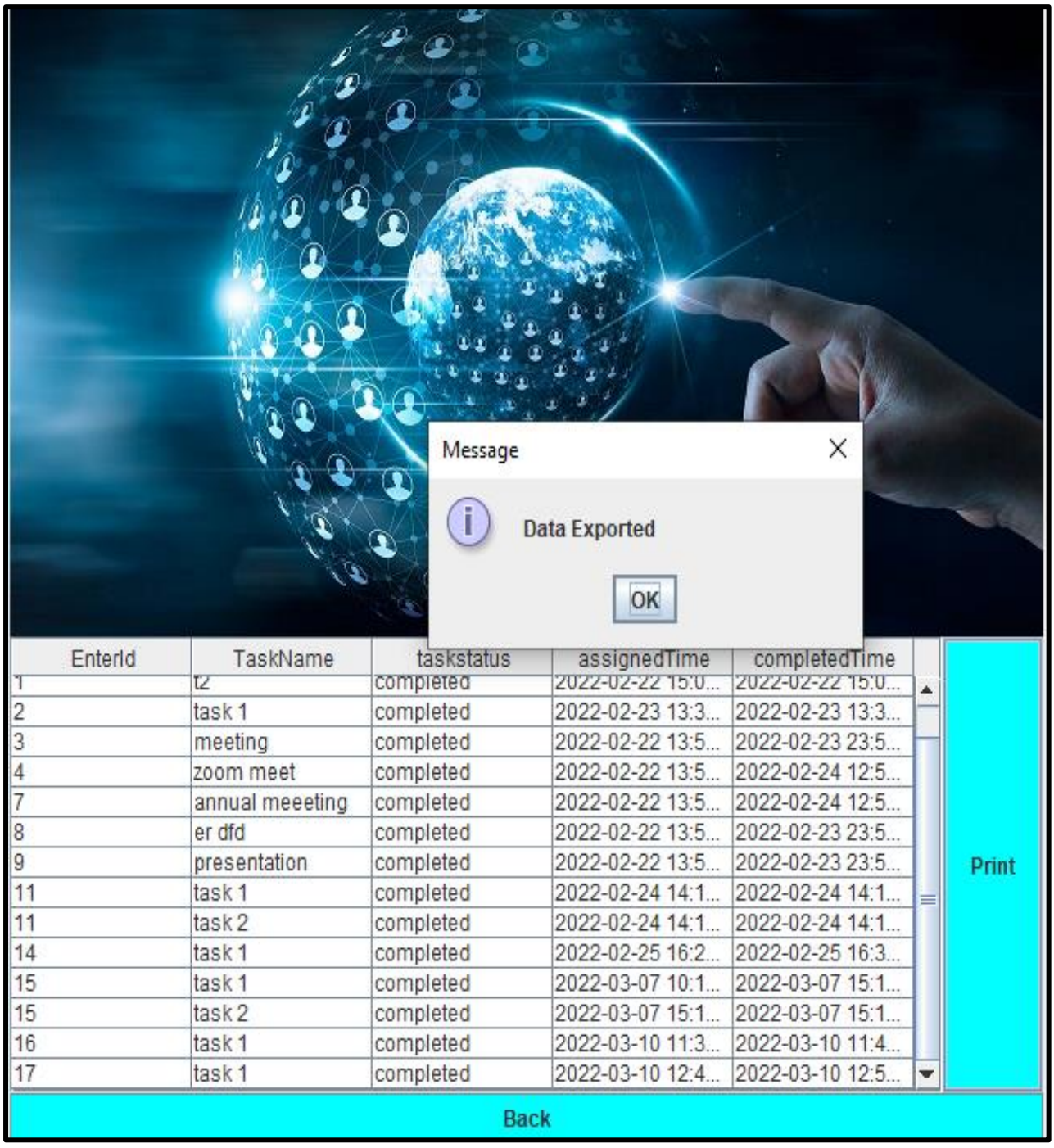
Print

Back

➢ UPDATE TASK

5.2) OUTPUT REPORTS WITH DATA

- **REPORT:**



5.3) SAMPLE PROGRAM CODE

➢ COSE USER LOGIN:

```
package com.CIS;

import java.awt.BorderLayout;

public class User extends JFrame {
    static User frame;
    private JPanel contentPane;

    /**
     * Launch the application.
     */
    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    ImagePanel2 panel2 = new ImagePanel2(new ImageIcon("C:\\Users\\Admin\\Downloads\\ta.jpg").getImage());
                    frame= new User();
                    frame.getContentPane().add(panel2);
                    frame.pack();
                    frame.setVisible(true);
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }

    /**
     * Create the frame.
     */
    public User() {
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setBounds(100, 100, 600, 400);
        contentPane = new JPanel();
        contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
        setContentPane(contentPane);


        JLabel lblTaskFacilitator = new JLabel("Task Facilitator");
```

```
        JLabel lblTaskFacilitator = new JLabel("Task Facilitator");
        lblTaskFacilitator.setFont(new Font("Tahoma", Font.PLAIN, 20));
        lblTaskFacilitator.setForeground(Color.WHITE);

        //JLabel imgLabel = new JLabel(new ImageIcon("C:\\\\Users\\\\Admin\\\\Downloads\\\\mainframeicon.png"));

        JButton btnAdminLogin = new JButton("Admin Login");
        btnAdminLogin.setBackground(Color.magenta);
        btnAdminLogin.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
            Adminlogin.main(new String[]{});
            frame.dispose();
            }
        });
        btnAdminLogin.setFont(new Font("Tahoma", Font.PLAIN, 15));

        JButton btnUserLogin = new JButton("User Login");
        btnUserLogin.setBackground(Color.yellow);
        btnUserLogin.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent arg0) {
                UserLogin.main(new String[]{});
            }
        });
        btnUserLogin.setFont(new Font("Tahoma", Font.PLAIN, 15));
        GroupLayout gl_contentPane = new GroupLayout(contentPane);
        gl_contentPane.setHorizontalGroup(
            gl_contentPane.createParallelGroup(Alignment.LEADING)
                .addGroup(gl_contentPane.createSequentialGroup()
                    .addGroup(gl_contentPane.createParallelGroup(Alignment.LEADING)
                        .addGroup(gl_contentPane.createSequentialGroup()
                            .addGap(64)
                            .addComponent(lblTaskFacilitator))
                        .addGroup(gl_contentPane.createSequentialGroup()
                            .addGap(140)
                            .addGroup(gl_contentPane.createParallelGroup(Alignment.TRAILING, false)
                                //.addComponent(imgLabel, Alignment.LEADING, GroupLayout.DEFAULT_SIZE, GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                                .addComponent(btnUserLogin, Alignment.CENTER, GroupLayout.DEFAULT_SIZE, GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                                .addComponent(btnAdminLogin, Alignment.CENTER, GroupLayout.DEFAULT_SIZE, 135, Short.MAX_VALUE))))
                    .addContainerGap(95, Short.MAX_VALUE))
```

```
        btnUserLogin.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent arg0) {
                UserLogin.main(new String[]{});
            }
        });
        btnUserLogin.setFont(new Font("Tahoma", Font.PLAIN, 15));
        GroupLayout gl_contentPane = new GroupLayout(contentPane);
        gl_contentPane.setHorizontalGroup(
            gl_contentPane.createParallelGroup(Alignment.LEADING)
                .addGroup(gl_contentPane.createSequentialGroup()
                    .addGroup(gl_contentPane.createParallelGroup(Alignment.LEADING)
                        .addGroup(gl_contentPane.createSequentialGroup()
                            .addGap(64)
                            .addComponent(lblTaskFacilitator))
                        .addGroup(gl_contentPane.createSequentialGroup()
                            .addGap(140)
                            .addGroup(gl_contentPane.createParallelGroup(Alignment.TRAILING, false)
                                //.addComponent(imgLabel, Alignment.LEADING, GroupLayout.DEFAULT_SIZE, GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                                .addComponent(btnUserLogin, Alignment.CENTER, GroupLayout.DEFAULT_SIZE, GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                                .addComponent(btnAdminLogin, Alignment.CENTER, GroupLayout.DEFAULT_SIZE, 135, Short.MAX_VALUE))))
                    .addContainerGap(95, Short.MAX_VALUE))
        );
        gl_contentPane.setVerticalGroup(
            gl_contentPane.createParallelGroup(Alignment.LEADING)
                .addGroup(gl_contentPane.createSequentialGroup()
                    .addContainerGap()
                    .addComponent(lblTaskFacilitator)
                    .addGap(32)
                    //.addComponent(imgLabel, GroupLayout.PREFERRED_SIZE, 182, GroupLayout.PREFERRED_SIZE)
                    .addGap(32)
                    .addComponent(btnAdminLogin, GroupLayout.PREFERRED_SIZE, 52, GroupLayout.PREFERRED_SIZE)
                    .addPreferredGap(ComponentPlacement.UNRELATED)
                    .addComponent(btnUserLogin, GroupLayout.PREFERRED_SIZE, 53, GroupLayout.PREFERRED_SIZE)
                    .addContainerGap(70, Short.MAX_VALUE))
        );
        contentPane.setLayout(gl_contentPane);

    }
}
```

CHAPTER 6 – DRAWBACK AND LIMITATIONS

- At time Admin can only one task assign to the user.
- More Man power
- Lack Of Security Of Data
- Time Consuming

CHAPTER 7 – PROPOSED ENHANCEMENTS

- The project has covered almost all the requirements.
- The project has a very vast scope in future.
- Project can updated in near future as and when requirement for the same arises, as it is very flexible in terms of expansion.
- Enhancements can done in an efficient manner.
- We can even update this particular software system with further modification establishment and can integrated with minimal modification.
- So project is flexible and can enhanced at any time with more advanced features.

CHAPTER 8 – CONCLUSION

- This project is developed successfully and the performance is found to be satisfactory. This project is designed to meet the requirements of assigning jobs. It has been developed in Java swing and the database has been built in My SQL server keeping in mind the specifications of the system.
- The admin will be able to add/register users for further task allocation using this Desktop application. The relationship between Admin and its users satisfy a good cordination to complete task allocating process.
- We have designed the project to provide the admin with easy retrieval of data, details of completed task. In this project, the admin is provided with an interface that can be used to add, view, delete users along with assigning various tasks and to track the task related details.
- To implement this as a desktop application we used the java swing framework technology. java has advantages such as enhanced performance, scalability, built- in security and simplicity. To build any desktop application using JAVA we have used java as frontend development and so on. MySQL was used as back-end database since it is one of the most popular open source databases, and it provides fast data access, easy installation and simplicity.

CHAPTER 9 – BIBILIOGRAPHY

Following books were helpful to us in building and understanding the concepts. These books proved to be great importance during the actual development i.e. Design & coding of the system.

1 – [HER 10] Herbert Schildt "The complete Reference Java" Tata McGraw hill, New Delhi, 2010

Bibliography/References/Glossary

https://www.tutorialspoint.com/index.htm

https://www.javatpoint.com

https://www.w3schools.com

https://html.com